

EXHIBIT 1

CONFIDENTIAL – OUTSIDE ATTORNEYS EYES ONLY

The Honorable James L. Robart

UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF WASHINGTON
AT SEATTLE

MOTOROLA MOBILITY, INC., and
GENERAL INSTRUMENT CORPORATION

Plaintiffs,

v.

MICROSOFT CORPORATION,

Defendant.

Civil Action No. C11-00343-JLR

PLAINTIFFS MOTOROLA
MOBILITY, INC. AND GENERAL
INSTRUMENT CORPORATION'S
RESPONSES TO DEFENDANT
MICROSOFT CORPORATION'S FIRST
SET OF INTERROGATORIES

PLAINTIFFS' RESPONSES TO DEFENDANT'S
FIRST SET OF INTERROGATORIES
CASE NO. C11-00343-JLR

ROPES & GRAY LLP
1900 UNIVERSITY AVENUE, 6TH FL.
EAST PALO ALTO, CA 94303
Telephone: (650) 617-4000
Fax: (650) 617-4090

CONFIDENTIAL – OUTSIDE ATTORNEYS EYES ONLY**INTERROGATORY NO. 11**

If you contend that any asserted claim of the Microsoft Patents-In-Suit is invalid, state for each claim the basis for this contention (including all facts, information, prior art and documents that You assert support or are pertinent to its contention of invalidity) including describing in detail where each element, limitation, or step of each claim is found in the alleged prior art (i.e., read each claim on the alleged prior art by providing, separately for each claim element, a claim- element by claim-element comparison of each claim to the structure, function or disclosure of the alleged prior art), and identifying the person or persons most knowledgeable about such contentions and bases.

RESPONSE TO INTERROGATORY NO. 11

Motorola objects to this Interrogatory as premature and to the extent that it seeks to impose any requirement or obligation on Motorola in addition to, beyond the scope of, or different from those imposed by the Federal Rules of Civil Procedure, Judge Robart's Standing Order for Patent Cases and/or Local Rules of the United States District Court for the Western District of Washington. Motorola also objects to this Interrogatory to the extent that it calls for the substance of expert opinion or anticipated testimony as the timing of such identification is governed by the Federal Rules of Civil Procedure, Judge Robart's Standing Order for Patent Cases and/or Local Rules of the United States District Court for the Western District of Washington. Motorola further objects to this interrogatory to the extent it seeks a legal conclusion and/or presents a question of law. Motorola also objects to this Interrogatory to the extent it seeks information protected by the attorney-client privilege, work product immunity, or other applicable privilege or immunity.

Subject to and without waiving its General Objections and foregoing specific objections, Motorola contends that each asserted claim of the Microsoft Patents-In-Suit is invalid under one or more sections of 35 U.S.C. §§ 102 or 103, and/or failure to comply with the requirements of 35 U.S.C. § 112. Motorola's invalidity contentions reflect present knowledge and contentions, and Motorola reserves all rights to modify and supplement these contentions without prejudice in the event that additional invalidity grounds are identified. Given that the parties have not yet specifically identified proposed terms for construction or provided their proposed constructions,

CONFIDENTIAL – OUTSIDE ATTORNEYS EYES ONLY

Motorola's invalidity contentions are necessarily preliminary. Motorola's contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way.

Motorola reserves the right to modify, amend and/or supplement these contentions in view of, without limitation, information provided by Microsoft concerning its infringement allegations; discovery concerning the alleged priority, conception, and reduction to practice dates for any of the asserted claims; additional prior art obtained through discovery or further investigation, including without limitation discovery from Microsoft or third parties; the Court's claim construction findings; or any other basis in law or in fact.

Motorola sets forth below an identification of references that Motorola may rely on to show that each asserted claim of the Microsoft Patents-In-Suit to be invalid, either alone or in combination with one or more references.

- Prior Art to U.S. Patent No. 6,339,780
 - U.S. Patent 6,584,498
 - U.S. Patent 5,528,744
 - U.S. Patent 6,377,978
 - U.S. Patent 5,768,539
 - U.S. Patent 5,886,683
 - U.S. Patent 6,816,880
 - U.S. Patent 6,182,072
 - U.S. Patent 5,978,848
 - U.S. Patent 5,960,435
 - U.S. Patent 6,108,673
 - U.S. Patent 5,995,756
 - U.S. Patent 6,011,537
 - U.S. Patent 6,487,588
 - U.S. Patent 6,437,758

PLAINTIFFS' RESPONSES TO DEFENDANT'S
FIRST SET OF INTERROGATORIES - 23
CASE NO. C11-00343-JLR

ROPES & GRAY LLP
1900 UNIVERSITY AVENUE, 6TH FL.
EAST PALO ALTO, CA 94303
Telephone: (650) 617-4000
Fax: (650) 617-4090

CONFIDENTIAL – OUTSIDE ATTORNEYS EYES ONLY

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

- U.S. Patent 5,896,444
- U.S. Patent 5,805,815
- U.S. Patent 5,790,785
- U.S. Patent 5,761,385
- U.S. Patent 6,493,002
- U.S. Patent 5,600,825
- U.S. Patent 5,572,643
- U.S. Publication No. 20030163431A1
- Heltzell, Dallas, *AOL users cheer blue bar's death*, Colorado Springs Gazette Telegraph, 30 June 1996
- Kasten, Alex S., *Off-computer CD-ROM and the game machines*, EMedia Professional, Vol. 10, No. 3, ISSN: 1090-946X, 1 March 1997
- Roulo, Mark, *Creating download progress bars for applets; Here's a way to keep users happy while they wait for your applet to download*, JavaWorld.com (Online), 1 December 1996
- Prior art to U.S. Patent 7,411,582
 - U.S. Patent 5,157,384
 - U.S. Patent 6,608,637
 - U.S. Patent 5,796,967
 - U.S. Patent 5,638,501
 - U.S. Patent 5,764,226
 - U.S. Patent 5,778,404
 - U.S. Patent 5,528,248
 - U.S. Patent 6,128,016
 - U.S. Patent 5,949,418
 - U.S. Patent 5,822,230
 - U.S. Patent 5,805,157
 - U.S. Patent 5,581,243
 - U.S. Patent 5,825,362
 - U.S. Patent 5,661,476
 - U.S. Patent 5,594,471

CONFIDENTIAL – OUTSIDE ATTORNEYS EYES ONLY

- U.S. Patent 5,606,702
- U.S. Patent 5,252,951
- U.S. Patent 5,509,103 (WO 9531264)
- U.S. Patent 6,611,258 (JP9190268)
- JP9251379
- Vanderheiden, G.C., *Cross disability assess to touch screen kiosks and ATMs*, Proceeding of 7th International Conference on Human Computer Interaction jointly with 13th Symposium on Human Interface, vol.1., pp. 417-20, Aug. 1997.
- Montaniz, F.1 and Mack, R.1, *A comparison of touch interface techniques for a graphical windowing software environment*, Proceedings of the Human Factors Society 35th Annual Meeting, 290-4 vol.1, 1991.
- Sternecker, G., *The Apple Graphics Tablet*, Creative Computing, v 7, n 1, 28-9, Jan. 1981.
- Tanaka, Toshinori and Kobayashi, Shunsuke, *Entry of data and command for an LCD by direct touch: an integrated LCD panel*, Digest of Technical Papers - SID International Symposium (Society for Information Display), v 17, p 318-320, 1986.
- William Buxton, Ralph Hill and Peter Rowley, *Issues and techniques in touch-sensitive tablet input*, SIGGRAPH '85 Proceedings of the 12th annual conference on Computer graphics and interactive techniques ACM New York, NY, USA ©1985.

INTERROGATORY NO. 12

Describe in detail all facts and circumstances that support or otherwise relate to Your claim for damages for alleged infringement of the General Instrument Patents-In-Suit, including, the demand for the patented product, the absence or presence of any and all allegedly non-infringing substitutes, the manufacturing and marketing capacity to exploit the demand, and the amount of profit You would have made and, for each such patent that You contend You are entitled to recover damages under the entire market value rule, state the economic connection between the invented feature and the accused products and all facts, documents, and/or witnesses You rely upon in support of its contention(s), and all facts, documents and/or witnesses You rely upon to show that the patent-related feature is the basis for customer demand.

RESPONSE TO INTERROGATORY NO. 12

Motorola objects to this Interrogatory as seeking information not relevant to any issue in this action and not reasonably calculated to lead to the discovery of admissible evidence.

Motorola also objects to this Interrogatory as premature and to the extent that it seeks to impose

CONFIDENTIAL – OUTSIDE ATTORNEYS EYES ONLY

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

As to objections,

Dated this 30th day of March, 2011

ROPES & GRAY LLP

By /s/ Gabrielle E. Higgins

Philip S. McCune, WSBA #21081
Lynn M. Engel, WSBA #21934
SUMMIT LAW GROUP PLLC
philm@summitlaw.com
lynne@summitlaw.com

Steven Pepe (pro hac vice)
Jesse J. Jenner (pro hac vice)
Anthony Pastor (pro hac vice)
Ropes & Gray LLP
1211 Avenue of the Americas
New York, NY 10036-8704
(212) 596-9046
jesse.jenner@ropesgray.com
steven.pepe@ropesgray.com
anthony.pastor@ropesgray.com

Norman H. Beamer (pro hac vice)
Gabrielle E. Higgins (pro hac vice)
Ropes & Gray LLP
1900 University Avenue, 6th Floor
East Palo Alto, CA 94303-2284
(650) 617-4030
norman.beamer@ropesgray.com
gabrielle.higgins@ropesgray.com

Paul M. Schoenhard (pro hac vice)
Ropes & Gray LLP
One Metro Center
700 12th Street NW, Suite 900
Washington, DC 20005-3948
(202) 508-4693
paul.schoenhard@ropesgray.com

**Attorneys for Plaintiffs Motorola Mobility,
Inc. and General Instrument Corporation**

CONFIDENTIAL – OUTSIDE ATTORNEYS EYES ONLY

VERIFICATION

Verification to follow.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

PLAINTIFFS' RESPONSES TO DEFENDANT'S
FIRST SET OF INTERROGATORIES - 29
CASE NO. C11-00343-JLR

ROPES & GRAY LLP
1900 UNIVERSITY AVENUE, 6TH FL.
EAST PALO ALTO, CA 94303
Telephone: (650) 617-4000
Fax: (650) 617-4090

CONFIDENTIAL – OUTSIDE ATTORNEYS EYES ONLY

CERTIFICATE OF SERVICE

I hereby certify that on this day I served a true and correct copy of PLAINTIFFS
MOTOROLA MOBILITY, INC. AND GENERAL INSTRUMENT CORPORATION'S
RESPONSES TO DEFENDANT MICROSOFT CORPORATION'S FIRST SET OF
INTERROGATORIES via electronic mail, on counsel of record below.

Arthur W. Harrigan, Jr., Esq.
Christopher T. Wion, Esq.
Shane P. Cramer, Esq.
Danielson, Harrigan, Leyh & Tollefson LLP
arthurh@dhlt.com
chrisw@dhlt.com
shanec@dhlt.com

David T. Pritikin, Esq.
Douglas I. Lewis, Esq.
John W. McBride, Esq.
Kevin C. Wheeler, Esq.
Richard A. Cederoth, Esq.
Sidley Austin LLP
bnester@sidley.com
dpritikin@sidley.com
dilewis@sidley.com
jwmcbride@sidley.com
kwheeler@sidley.com
rcederoth@sidley.com

DATED this 30th day of March, 2011.

/s/ Matthew R. Clements

Matthew R. Clements

PLAINTIFFS' RESPONSES TO DEFENDANT'S
FIRST SET OF INTERROGATORIES - 30
CASE NO. C11-00343-JLR

ROPES & GRAY LLP
1900 UNIVERSITY AVENUE, 6TH FL.
EAST PALO ALTO, CA 94303
Telephone: (650) 617-4000
Fax: (650) 617-4090

EXHIBIT 2

HONORABLE JAMES L. ROBART

IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF WASHINGTON
AT SEATTLE

MICROSOFT CORPORATION,

Plaintiff,

v.

MOTOROLA, INC., et al.,

Defendants.

No. C10-1823-JLR

MOTOROLA MOBILITY, INC., et al.,

Plaintiffs,

v.

MICROSOFT CORPORATION,

Defendant.

DEFENDANT AND CROSS-PLAINTIFF MICROSOFT CORPORATION'S
ASSERTED CLAIMS AND PRELIMINARY INFRINGEMENT CONTENTIONS

Defendant Microsoft Corporation (“Microsoft”) provides to Plaintiffs Motorola Mobility, Inc.’s (“Motorola Mobility”), Motorola Solutions, Inc. (“Motorola Solutions”) and General Instrument Corporation’s (“General Instrument”) (collectively, “Motorola”) the following Asserted Claims and Preliminary Infringement Contentions in compliance with the Court’s Standing Order for Patent Cases and with the Minute Order Setting Trial Dates and Related Dates dated August 5, 2011. Microsoft reserves the right to amend this disclosure to conform to the results of ongoing discovery.

(1) The identity of each claim alleged to be infringed

U.S. Patent No. 6,339,780 (“the ‘780 Patent”): Claims 1-6, 9-14, 17, 18, 20, 21, and 32-42.

U.S. Patent No. 7,411,582 (“the ‘582 Patent”): Claims 1-4, 6, 8-11, 13-23, 25-31.

These claims will be hereinafter referred to as the “Microsoft Asserted Claims.” Microsoft’s investigation as to Motorola’s infringement is ongoing. Accordingly, Microsoft reserves the right to assert more or fewer claims based on further information that may become available during ongoing discovery.

(2) The identity of Motorola’s accused devices by specific name and model number for each asserted claim

Motorola Mobility is currently accused of infringing some or all of the Microsoft Asserted Claims by making, using, selling, offering to sell, and/or importing the following products, hereinafter referred to as the Accused Motorola Devices:

The ‘780 Patent	The ‘582 Patent
Atrix ME860, MB861	Atrix ME860, MB861
Bravo MB520	Bravo MB520

The '780 Patent	The '582 Patent
Charm MB502, ME502	
	Citrus WX445
	Cliq XT MB501
Cliq 2 MB611	Cliq 2 MB611
Defy MB525	Defy MB525
Devour A555	
Droid A855	Droid A855
Droid 2 A955	Droid 2 A955
Droid 2 Global A956	Droid 2 Global A956
Droid 3 XT862	Droid 3 XT862
Droid Pro XT610	Droid Pro XT610
Droid X MB810	Droid X MB810
Droid X2 MB870	Droid X2 MB870
Flipout MB511	Flipout MB511
Flipside MB508	Flipside MB508
	i1
Photon 4G MB855	Photon 4G MB855
Spice XT300	
Titanium	Titanium
Triumph WX435	Triumph WX435

The '780 Patent	The '582 Patent
XPRT MB612	
Xoom MZ600, MZ601, MZ603, MZ604, MZ605, MZ606 (except as to Claims 6, 33, and 39).	

Microsoft's investigation as to Motorola's infringement is ongoing. Accordingly, Microsoft reserves the right to amend the list of Accused Motorola Devices based on further information that may become available during ongoing discovery.

(3) A chart that identifies specifically where each element of each asserted claim is found within each accused device/method/etc.

Microsoft accuses Motorola Mobility of having infringed, induced the infringement of and/or contributorily infringed, and continuing to infringe, induce infringement of and/or contributorily infringe the Microsoft Asserted Claims pursuant to 35 U.S.C. § 271 (a), (b) and/or (c), literally or under the doctrine of equivalents, in the United States, by making, using, selling, offering to sell and/or importing the Accused Motorola Devices. Infringement charts for the '780 patent, containing evidence of direct and indirect infringement, are attached as Exhibit A. Infringement charts for the '582 patent, containing evidence of direct and indirect infringement, are attached as Exhibit B.

(4) Whether each asserted claim is literally or equivalently infringed

Microsoft reserves the right to assert infringement under the doctrine of equivalents for any limitation of the '780 or '582 patents that Motorola contends is not literally present in its

1 Accused Devices. Microsoft further reserves the right to assert infringement under the doctrine
2 of equivalents based on a claim construction ruling on any disputed claim terms and/or based
3 on further information that may become available during ongoing discovery.

4
5 **(5) The priority date to which each asserted claim allegedly is entitled**

6 The '780 Patent: The priority date for each claim of the '780 patent asserted in Section
7 (2) above is the earliest effective filing date on the face of the patent: May 6, 1997.

8 The '582 Patent: The priority date for each claim of the '582 patent asserted in Section
9 (2) above is the priority date of U.S. Patent Application Ser. No 08/991,277, to which the
10 patent claims priority ('582 patent, 1:6-8): December 16, 1997.

11 Microsoft's investigation as to conception of the inventions disclosed in both the '582
12 and '780 patents is ongoing. Accordingly, Microsoft reserves the right to amend this priority
13 date based on further information that may become available during ongoing discovery.

14
15
16 DATED this 2nd day of September, 2011.

17 DANIELSON HARRIGAN LEYH & TOLLEFSON LLP

18 By /s/ Arthur W. Harrigan, Jr.

19 Arthur W. Harrigan, Jr., WSBA #1751

20 Christopher Wion, WSBA #33207

21 Shane P. Cramer, WSBA #35099

22 T. Andrew Culbert, WSBA #35925

23 David E. Killough, WSBA #40185

24 MICROSOFT CORPORATION

25 1 Microsoft Way

Redmond, WA 98052

Phone: 425-882-8080

Fax: 425-869-1327

1 David T. Pritikin, *Pro Hac Vice*
2 Richard A. Cederoth, *Pro Hac Vice*
3 Douglas I. Lewis, *Pro Hac Vice*
4 John W. McBride, *Pro Hac Vice*
5 SIDLEY AUSTIN LLP
6 One South Dearborn
7 Chicago, IL 60603
8 Phone: 312-853-7000
9 Fax: 312-853-7036

10 Brian R. Nester, *Pro Hac Vice*
11 SIDLEY AUSTIN LLP
12 1501 K Street NW
13 Washington, DC 20005
14 Telephone: 202-736-8000
15 Fax: 202-736-8711

16 Counsel for Microsoft Corporation
17
18
19
20
21
22
23
24
25

CERTIFICATE OF SERVICE

I hereby certify that on September 2, 2011 I served a true and correct copy of Defendant and Cross-Plaintiff Microsoft Corporation's Asserted Claims and Preliminary Infringement Contentions via electronic mail on the counsel of record below.

Attorneys for Defendants Motorola Solutions, Inc., Motorola Mobility, Inc., and General Instrument Corporation

Philip S. McCune
Lynn M. Engle
Summit Law Group

Steven Pepe
Jesse J. Jenner
Norman Beamer
Paul M. Schoenhard
Ropes & Gray

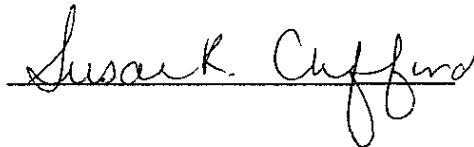


EXHIBIT 3

The Honorable James L. Robart

UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF WASHINGTON
AT SEATTLE

MICROSOFT CORPORATION,

Plaintiff,

v.

MOTOROLA, INC., et al.,

Defendants.

Civil Action No. C10-01823-JLR

PLAINTIFFS MOTOROLA
MOBILITY, INC. AND GENERAL
INSTRUMENT CORPORATION'S
SUPPLEMENTAL RESPONSE TO
DEFENDANT MICROSOFT
CORPORATION'S
INTERROGATORY NO. 11

MOTOROLA MOBILITY, INC., et al.

Plaintiffs,

v.

MICROSOFT CORPORATION,

Defendant.

PLAINTIFFS'SUPPLMENTAL RESPONSE TO
DEFENDANT'S INTERROGATORY NO. 11
CASE NO. C10-01823-JLR

ROPES & GRAY LLP

1211 AVENUE OF THE AMERICAS

NEW YORK, NY 10036

Telephone: (212) 596-9000

Fax: (212) 596-9090

Pursuant to Rules 26 and 33 of the Federal Rules of Civil Procedure, Local Patent Rule 121, and the Amended Order Setting Trial Dates and Related Dates (Dkt. No. 93), Motorola Mobility, Inc. ("Motorola Mobility") and General Instrument Corporation ("General Instrument") (collectively, "Motorola") provide a supplemental response and objections to Microsoft Corporation's ("Microsoft") First Set of Interrogatories, Interrogatory No. 11 as follows:

GENERAL OBJECTIONS

Motorola hereby incorporates the General Objections set forth in its March 30, 2011 Responses to Defendant Microsoft Corporation's First Set of Interrogatories.

OBJECTIONS TO DEFINITIONS AND INSTRUCTIONS

Motorola hereby incorporates the Objections to Definitions and Instructions set forth in its March 30, 2011 Responses to Defendant Microsoft Corporation's First Set of Interrogatories.

RESPONSES TO INTERROGATORIES

INTERROGATORY NO. 1

If you contend that any asserted claim of the Microsoft Patents-In-Suit is invalid, state for each claim the basis for this contention (including all facts, information, prior art and documents that You assert support or are pertinent to its contention of invalidity) including describing in detail where each element, limitation, or step of each claim is found in the alleged prior art (i.e., read each claim on the alleged prior art by providing, separately for each claim element, a claim- element by claim-element comparison of each claim to the structure, function or disclosure of the alleged prior art), and identifying the person or persons most knowledgeable about such contentions and bases.

FIRST SUPPLEMENTAL RESPONSE TO INTERROGATORY NO. 1

Motorola objects to this Interrogatory as premature and to the extent that it seeks to impose any requirement or obligation on Motorola in addition to, beyond the scope of, or different from those imposed by the Federal Rules of Civil Procedure, Judge Robart's Standing Order for Patent Cases and/or Local Rules of the United States District Court for the Western District of Washington. Motorola also objects to this Interrogatory to the extent that it calls for

1 the substance of expert opinion or anticipated testimony as the timing of such identification is
2 governed by the Federal Rules of Civil Procedure, Judge Robart's Standing Order for Patent
3 Cases and/or Local Rules of the United States District Court for the Western District of
4 Washington. Motorola further objects to this interrogatory to the extent it seeks a legal
5 conclusion and/or presents a question of law. Motorola also objects to this Interrogatory to the
6 extent it seeks information protected by the attorney-client privilege, work product immunity, or
7 other applicable privilege or immunity.

8 Subject to and without waiving its General Objections and foregoing specific objections,
9 Motorola contends that each asserted claim of the Microsoft Patents-In-Suit is invalid under one
10 or more sections of 35 U.S.C. §§ 102 or 103, and/or failure to comply with the requirements of
11 35 U.S.C. § 112. This case is still in the early stages of discovery. Motorola has not yet
12 completed its investigation, collection of information, discovery, or analysis related to this
13 action. Motorola's invalidity contentions reflect present knowledge and contentions, and
14 Motorola reserves all rights to modify and supplement these contentions without prejudice as
15 necessary based on further discovery and investigation, review of newly or yet-to-be produced
16 documents, or any rulings of the Court. Given that the parties have not yet specifically
17 identified proposed terms for construction or provided their proposed constructions, Motorola's
18 invalidity contentions are necessarily preliminary. Motorola's contentions herein are not, and
19 should in no way be seen as, admissions or adoptions as to any particular claim scope or
20 construction, or as any admission that any particular element is met in any particular way.
21 Motorola objects to any attempt to imply claim constructions from any identification of
22 potential prior art.

23 Motorola reserves the right to modify, amend and/or supplement these contentions in
24 view of, without limitation, information provided by Microsoft concerning its infringement
25 allegations; discovery concerning the alleged priority, conception, and reduction to practice
26 dates for any of the asserted claims; additional prior art obtained through discovery or further

1 investigation, including without limitation discovery from Microsoft or third parties; the Court's
 2 claim construction findings; or any other basis in law or in fact.

3 Motorola sets forth below an identification of references that Motorola may rely on to
 4 show each asserted claim of the Microsoft Patents-In-Suit to be invalid, either alone or in
 5 combination with one or more references. Motorola further provides in the attached Exhibits
 6 (Exhibits A-K) exemplar charts containing detailed descriptions illustrative of where each
 7 element of each asserted claim of the Microsoft Patents-In-Suit is anticipated or rendered
 8 obvious. Each of the exemplar charts sets forth where each charted reference anticipates the
 9 asserted claims of the Microsoft Patents-In-Suit. Obviousness arguments are set forth in the
 10 alternative where appropriate. The charted prior art references may contain additional support
 11 for particular claim limitations. Motorola expressly reserves the right to rely on uncited portions
 12 of those prior art references, other documents, and expert testimony to provide context for or to
 13 aid in understanding the cited portions of the references. Where Motorola cites to a particular
 14 figure in a reference, the citation should be understood to encompass the caption and description
 15 of the figure and any text relating to or discussing the figure. Conversely, where Motorola cites
 16 to a particular text referring to a figure, the citation should be understood to include the
 17 referenced figure as well. In the Exhibits attached, Motorola provides citations of where the
 18 prior art references disclose subject matter recited in preambles, without regard to whether the
 19 preambles are limitations of the claims. Motorola reserves the right to argue that the preambles
 20 are or are not limitations.

21 The claims of the Microsoft Patents-In-Suit do not represent a patentable advance over
 22 the prior art, and accordingly are anticipated by and/or obvious, taken alone or in combination,
 23 in view of the following references:

- 24 • Prior Art to U.S. Patent No. 6,339,780
- 25 • U.S. Patent 6,584,498 (06/24/2003)
- 26 • U.S. Patent 5,528,744 (06/18/1996)

PLAINTIFFS'SUPPLMENTAL RESPONSE TO
 DEFENDANT'S INTERROGATORY NO. 11
 CASE NO. C10-01823-JLR

ROPES & GRAY LLP
 1211 AVENUE OF THE AMERICAS
 NEW YORK, NY 10036
 Telephone: (212) 596-9000
 Fax: (212) 596-9090

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

- U.S. Patent 6,377,978 (04/23/2002)
- U.S. Patent 5,768,539 (06/16/1998)
- U.S. Patent 5,886,683 (03/23/1999)
- U.S. Patent 6,816,880 (11/09/2004)
- U.S. Patent 6,182,072 (01/30/2001)
- U.S. Patent 5,978,848 (11/02/1999)
- U.S. Patent 5,960,435 (09/28/1999)
- U.S. Patent 6,108,673 (08/22/2000)
- U.S. Patent 5,995,756 (11/30/1999)
- U.S. Patent 6,011,537 (01/04/2000)
- U.S. Patent 6,487,588 (11/26/2002)
- U.S. Patent 6,437,758 (08/20/2002)
- U.S. Patent 5,896,444 (04/20/1999)
- U.S. Patent 5,805,815 (09/08/1999)
- U.S. Patent 5,790,785 (08/04/1998)
- U.S. Patent 5,761,385 (06/02/1998)
- U.S. Patent 6,493,002 (12/10/2002)
- U.S. Patent 5,600,825 (02/04/1997)
- U.S. Patent 5,572,643 (11/05/1996)
- U.S. Patent 6,023,698 (02/08/2000)
- U.S. Patent 5,737,619 (04/07/1998)
- U.S. Patent 6,263,507 (07/17/2001)
- U.S. Patent 5,513,126 (04/30/1996)
- U.S. Patent 5,515,496 (05/07/1996)
- WO 98/25198 (06/11/1998)
- U.S. Publication No. 20030163431A1 (08/28/2003)
- Heltzell, Dallas, *AOL users cheer blue bar's death*, Colorado Springs Gazette Telegraph, 30 June 1996.
- Kasten, Alex S., *Off-computer CD-ROM and the game machines*, EMedia Professional, Vol. 10, No. 3, ISSN: 1090-946X, 1 March 1997.

PLAINTIFFS'SUPPLMENTAL RESPONSE TO
DEFENDANT'S INTERROGATORY NO. 11
CASE NO. C10-01823-JLR

ROPES & GRAY LLP
1211 AVENUE OF THE AMERICAS
NEW YORK, NY 10036
Telephone: (212) 596-9000
Fax: (212) 596-9090

- 1 • Roulo, Mark, *Creating download progress bars for applets; Here's a way to*
2 *keep users happy while they wait for your applet to download*, JavaWorld.com
(Online), 1 December 1996.
- 3 • S. Gessler and A. Kotulla. PDAs as Mobile WWW Browsers. *Computer*
4 *Networks and ISDN Systems* 28 (1995) 53-59.
- 5 • M. Liljeberg, T. Alanko et al. Optimizing World-Wide Web for Weakly
6 Connected Mobile Workstations: An Indirect Approach. *Proceedings of SDNE*
7 '95 (June 5-6 Whistler, Canada).
- 8 • M. Liljeberg, H. Helin, et al. Enhanced Services for World Wide Web in Mobile
9 WAN Environment. University of Helsinki Report C-1996-28 (Helsinki, April
10 1996).
- 11 • M. Liljeberg, H. Helin, et al. MOWGLI WWW Software: Improved Usability of
12 WWW in Mobile WAN Environments.
- 13 • I. Cooper and R. Shufflebotham. PDAWeb Browsers: Implementation Issues.
14 (University of Kent at Canterbury, November 9, 1995).
- 15 • M. Pichler, G. Orasche, and K. Andrews. VRweb: A Multi-System VRML
16 Viewer. (San Diego, California, December 1995).
- 17 • B. Gaines. Supporting Collaboration Through Multimedia Digital Archives.
18 Knowledge Science Institute, University of Calgary. (Version 1.0, November
19 1994).
- 20 • B. Kelly. Running a World Wide Web Service. Advisory Group on Computer
21 Graphics. (University of Leeds, February 1995).
- 22 • J. Bartlett. Experience with a Wireless World Wide Web Client. IEEE
23 COMPCON 95 (San Francisco, March 1995).
- 24 • G. Voelker and B. Bershad. Mobisaic: An Information System for a Mobile
25 Wireless Computing Environment. Workshop on Mobile Computing Systems
26 and Applications (Santa Cruz, December 1994).
- T. Watson. Application Design for Wireless Computing. 1994 Mobile
Computing Systems and Applications Workshop Position Paper (August 1994).
- NetHopper Version 3.0 User's Manual. 1997 AllPen Software, Inc.
- NetHopper Version 3.2 User's Manual. 1997 AllPen Software, Inc.
- NetHopper 2.0. Pen Lab Review. Pen Computing Issue #11. (July/August 1996).
- PocketWeb WWW Browser for the Apple Newton MessagePad. TecO.
<http://www.teco.edu/pocketweb/>
- R. Simon. The Newton in a Windows Environment. Worldwide Newton
Conference. (San Francisco, January 2006).
- A. Iellimo, Jr. PDAs the Next Generation. *Network World* (March 13, 1995).

- B. Meyers. A Taxonomy of Window Manager User Interfaces. IEEE. (September 1988).
- T. Sundsted. Java and event handling. JAVAWorld. (August 1, 1996).
- T. Sundsted. Introduction to the AWT. JAVAWorld. (July 1, 1996).
- T. Bickmore and B. Schilit. Digestor: Device-independent Access to the World Wide Web. (FX Palo Alto Laboratory).
- T. Kamba, S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. (1996). "Using small screen space more efficiently."
- M. Lauff and H. Gellerson. Multimedia Client Implementation on Personal Digital Assistants. TecO.
- E. Morgan. Implementing TCP/IP communications with HyperCard. Information Technology and Libraries 11(4):421-432 (December 1992).
- B. Meyers. The Importance of Percent-Done Progress Indicators for Computer-Human Interfaces. CHI '85 Proceedings (April 1985).
- A. deLespinasse. Rover Mosaic: E-mail Communication for a Full-Function Web Browser. Massachusetts Institute of Technology. June 1995.
- R. Robertson. Binary Communications. Newton Technology Journal. March 1997.
- Newton Programmer's Guide For Newton 2.0. Apple Press. Addison-Wesley Publishing Company. 1996.
- G. Macisaac. Priming the Cognitive Pump: Implicit Memory and Navigating Multiple Window Interfaces. The University of British Columbia. October 1994.
- F. Ergen. Effects of Interface Format, Feedback Style, and System Lag On the Usability of Hand-Held Internet Controllers. Virginia Polytechnic Institute and State University. December 17, 1996.
- Mosaic for Windows: A Hands-On Configuration and Set-Up Guide to Popular Web Browsers. Springer-Verlag New York, Inc. 1995.
- Web Browsers (e.g., NetHopper, Newt's Cape, PocketWeb, Vogon, etc.)
- Prior art to U.S. Patent 7,411,582
 - U.S. Patent 4,899,136 (02/06/1990)
 - U.S. Patent 5,157,384 (10/20/1992)
 - U.S. Patent 6,608,637 (08/19/2003)
 - U.S. Patent 5,796,967 (08/18/1998)
 - U.S. Patent 5,638,501 (06/10/1997)

- 1 • File wrapper to U.S. Patent 5,638,501
- 2 • U.S. Patent 5,764,226 (06/09/1998)
- 3 • U.S. Patent 5,778,404 (07/07/1998)
- 4 • U.S. Patent 5,528,248 (06/18/1996)
- 5 • U.S. Patent 6,128,016 (10/03/2000)
- 6 • U.S. Patent 5,949,418 (09/07/1999)
- 7 • U.S. Patent 5,822,230 (10/13/1998)
- 8 • U.S. Patent 5,805,157 (09/08/1998)
- 9 • U.S. Patent 5,581,243 (5,581,243)
- 10 • U.S. Patent 5,825,362 (10/20/1998)
- 11 • U.S. Patent 5,661,476 (08/26/1997)
- 12 • U.S. Patent 5,594,471 (01/14/1997)
- 13 • U.S. Patent 5,606,702 (02/25/1997)
- 14 • U.S. Patent 5,252,951 (10/12/1993)
- 15 • File wrapper to U.S. Patent 5,252,951
- 16 • U.S. Patent 5,509,103 (WO 9531264) (04/16/1996)
- 17 • U.S. Patent 6,611,258 (JP9190268) (08/26/2003)
- 18 • U.S. Patent 5,454,046 (09/26/1995)
- 19 • U.S. Patent 5,455,901 (10/03/1995)
- 20 • U.S. Patent 5,371,844 (12/06/1994)
- 21 • U.S. Patent 5,911,485 (06/15/1999)
- 22 • U.S. Patent 5,717,425 (02/10/1998)
- 23 • U.S. Patent 5,812,117 (09/22/1998)
- 24 • U.S. Patent 5,603,053 (02/11/1997)
- 25 • U.S. Patent 5,689,669 (11/18/1997)
- 26 • U.S. Patent 5,611,031 (03/11/1997)
- U.S. Patent 5,479,536 (12/26/1995)
- U.S. Patent 5,250,929 (10/05/1993)
- U.S. Patent 5,583,539 (12/10/1996)
- U.S. Patent 5,148,155 (09/15/1992)
- Filewrapper to U.S. Patent 5,148,155

PLAINTIFFS'SUPPLMENTAL RESPONSE TO
DEFENDANT'S INTERROGATORY NO. 11
CASE NO. C10-01823-JLR

ROPES & GRAY LLP
1211 AVENUE OF THE AMERICAS
NEW YORK, NY 10036
Telephone: (212) 596-9000
Fax: (212) 596-9090

- U.S. Patent 5,260,697 (11/09/1993)
- U.S. Patent 6,154,209 (11/28/2000)
- U.S. Patent 4,725,694 (02/16/1988)
- WO 9,532,469 (11/30/95)
- GB 2,309,365 (07/23/97)
- EP 0,464,712 (01/18/92)
- EP 0,811,940 (10/12/97)
- WO 9,415,273 (07/07/94)
- JP 7,098,640 A2 (04/11/95)
- JP 9,251,379 (09/22/97)
- Vanderheiden, G.C., Cross disability assess to touch screen kiosks and ATMs, Proceeding of 7th International Conference on Human Computer Interaction jointly with 13th Symposium on Human Interface, vol.1. pp. 417-20, Aug. 1997.
- Montaniz, F.1 and Mack, R.1, A comparison of touch interface techniques for a graphical windowing software environment, Proceedings of the Human Factors Society 35th Annual Meeting, 290-4 vol.1, 1991.
- Sternecker, G., The Apple Graphics Tablet, Creative Computing, v 7, n 1, 28-9, Jan. 1981.
- Tanaka, Toshinori and Kobayashi, Shunsuke, Entry of data and command for an LCD by direct touch: an integrated LCD panel, Digest of Technical Papers - SID International Symposium (Society for Information Display), v 17, p 318-320, 1986.
- William Buxton, Ralph Hill and Peter Rowley, Issues and techniques in touch-sensitive tablet input, SIGGRAPH '85 Proceedings of the 12th annual conference on Computer graphics and interactive techniques ACM New York, NY, USA ©1985.
- Brown, E., Buxton, W. & Murtagh, K., Windows on tablets as a means of achieving virtual input devices, In D. Diaper et al. (Eds), Human-Computer Interaction - INTERACT '90. Amsterdam: Elsevier Science Publishers B.V. (North-Holland), 675-681, 1990.
- Apple Press, Apple Computer Inc., Newton's Programmer's Guide, For Newton 2.0, Addison Wesley Publishing Company (1996).
- Newton – Apple MessagePad Handbook (Apple) 1995.
- The NewtonScript Programming Language (Apple) 1996.
- Apple Press, Apple Computer Inc., Newton 2.0 User Interface Guidelines. Addison Wesley Publishing Company (1996).

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

- Apple Press, Apple Computer Inc., Newton's Programmer's Reference. For Newton 2.0. The Apple Publishing System (1996)
- Taligent Keeps Its Promises, Oct. 23, 1995, Information Week
- Simon –Mobile Communications Made Simple, (Bell South, 1994)
- Simonizing the PDA – BellSouth's communicative Simon is a milestone in the evolution of the PDA., Dec. 1994
- Magic Cap-A Guide to General Magic's Revolutionary Communicator Software by Barbara Knaster; Addison-Wesley Publishing Company (February 1994)
- Eliezer Kantorowitz, Oded Sudarsky, The Adaptable Use Interface, Computer Practices Communication of the ACM- Vol. 32 Number 11,1352-1358, Nov. 1989
- Peter Y.F. Wu, Research Report – User Interface Management Systems: Survey and Assessment (IBM) May 22, 1989
- Magic Link User's Guide PIC-1000 - Personal Intelligent Communicator (Sony) 1994
- DATAMAN – The Complete Contact Management Solution for the Newton OS- PMI and StandAlone Inc. 1997
- Joel F. Bartlett- Experience With A Wireless World Wide Web Client- WRL Technical Note TN-46, Digital Western Research Laboratory, March 1995
- Albert Iellimo, PDAs The Next Generation Network World 3/13/1995
- Liu, Raymond et al. "A Platform For Multilingual Application Development with Unicond Standard." International Conference on Chinese Computing '94 (ICCC94). The Latest Technological Advancement and Application. June, 1-4, 1994. Singapore.
- Bartlett W., Mel et al. "Tablet: Personal Computer in the Year 2000." Communication of the ACM . Vol. 31.6. June, 1988. 639-646.
- Want, Roy et al. "The ParcTab Ubiquitous Computing Experiment" (March 1995)
- Brockschmidt, Kraig. Inside Ole, Second Edition. Microsoft Press. (1995)

As to objections,

Dated this 30th day of September, 2011

ROPES & GRAY LLP

By /s/ Steven Pepe

Philip S. McCune, WSBA #21081
Lynn M. Engel, WSBA #21934
SUMMIT LAW GROUP PLLC
philm@summitlaw.com
lynne@summitlaw.com

Steven Pepe (*pro hac vice*)
Jesse J. Jenner (*pro hac vice*)
Ropes & Gray LLP
1211 Avenue of the Americas
New York, NY 10036-8704
(212) 596-9046
jesse.jenner@ropesgray.com
steven.pepe@ropesgray.com

Norman H. Beamer (*pro hac vice*)
Gabrielle E. Higgins (*pro hac vice*)
Ropes & Gray LLP
1900 University Avenue, 6th Floor
East Palo Alto, CA 94303-2284
(650) 617-4030
norman.beamer@ropesgray.com
gabrielle.higgins@ropesgray.com

Paul M. Schoenhard (*pro hac vice*)
Ropes & Gray LLP
One Metro Center
700 12th Street NW, Suite 900
Washington, DC 20005-3948
(202) 508-4693
paul.schoenhard@ropesgray.com

**Attorneys for Plaintiffs Motorola Mobility,
Inc. and General Instrument Corporation**

PLAINTIFFS'SUPPLEMENTAL RESPONSE TO
DEFENDANT'S INTERROGATORY NO. 11
CASE NO. C10-01823-JLR

ROPES & GRAY LLP
1211 AVENUE OF THE AMERICAS
NEW YORK, NY 10036
Telephone: (212) 596-9000
Fax: (212) 596-9090

VERIFICATION

Verification to follow.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

PLAINTIFFS'SUPPLMENTAL RESPONSE TO
DEFENDANT'S INTERROGATORY NO. 11
CASE NO. C10-01823-JLR

ROPES & GRAY LLP
1211 AVENUE OF THE AMERICAS
NEW YORK, NY 10036
Telephone: (212) 596-9000
Fax: (212) 596-9090

CERTIFICATE OF SERVICE

I hereby certify that on this day I served a true and correct copy of PLAINTIFFS
MOTOROLA MOBILITY, INC. AND GENERAL INSTRUMENT CORPORATION'S
SUPPLEMENTAL RESPONSE TO DEFENDANT MICROSOFT CORPORATION'S
INTERROGATORY NO. 11 via electronic mail, on counsel of record below.

Arthur W. Harrigan, Jr., Esq.
Christopher T. Wion, Esq.
Shane P. Cramer, Esq.
Danielson, Harrigan, Leyh & Tollefson LLP
arthurh@dhlt.com
chrisw@dhlt.com
shanec@dhlt.com

David T. Pritikin, Esq.
Douglas I. Lewis, Esq.
John W. McBride, Esq.
Kevin C. Wheeler, Esq.
Richard A. Cederoth, Esq.
Sidley Austin LLP
bnester@sidley.com
dpritikin@sidley.com
dilewis@sidley.com
jwmcbride@sidley.com
kwheeler@sidley.com
rcederoth@sidley.com

DATED this 30th day of September, 2011.

/s/ Michael P. Duffey

Michael P. Duffey

EXHIBIT A

Exemplar Chart of U.S. Patent 6,339,780**U.S. Patent 5,528,744 (“Vaughton ‘744’”)¹
Claims 1-6, 9-14, 17, 18, 20, 21 and 32-42**

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
1	1. A hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area,	<p>“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8. One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link.</p> <p>“Such systems provide a number of display areas (windows) whose size, shape and position within the display may be manipulated by the user.” Col. 1, ll. 37-39.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser on an information processing device having a limited display area.</p> <p>In addition, the combination of Vaughton ‘744 and one or more references describing handheld devices with browsers renders this claim obvious. Gessler², Cooper³,</p>

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Data Processing System” issued to Vaughton; filed August 30, 1995; issued June 18, 1996.

² S. Gessler and A. Kotulla. PDAs as Mobile WWW Browsers. Computer Networks and ISDN Systems 28 (1995) 53-59 (“Gessler”) (see pgs. 53 and 56-58).

³ I. Cooper and R. Shufflebotham. PDAWeb Browsers: Implementation Issues. (University of Kent at Canterbury, November 9, 1995) (“Cooper”) (see pgs. 3 and 5).

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
		Bartlett ⁴ , Watson ⁵ , Kamba ⁶ , and Lauff ⁷ each teach a hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area (e.g., a PDA, handheld, mobile device, etc.), and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. ⁸ One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Additionally, Bartlett cites Gessler and Cooper cites Gessler and Bartlett. Moreover, all of these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.1	wherein the hypermedia browser has a content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing area during times when the browser is loading content,	“The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows, wherein execution of a first task may trigger execution of a second task characterised in that when execution of said first task triggers execution of said second task, said processor uses data identifying said first task’s display window(s) and to control display of said second task’s display window(s) so as to be connected to and move with said first task’s display

⁴ J. Bartlett. Experience with a Wireless World Wide Web Client. IEEE COMPCON 95 (San Francisco, March 1995) (“Bartlett”) (see pgs. 1, 4 and 5).

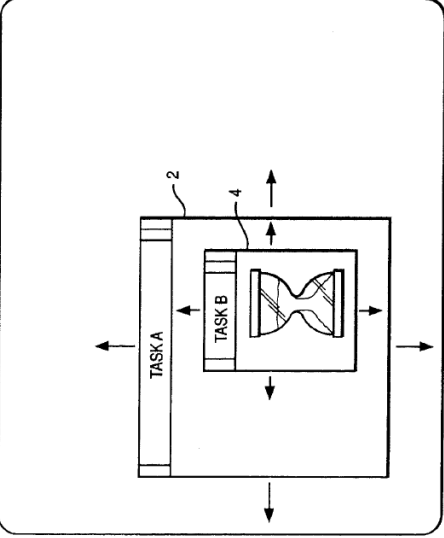
⁵ T. Watson. “Application Design for Wireless Computing” (1994 Mobile Computing Systems and Applications Workshop Position Paper (August 1994)).

⁶ T. Kamba, S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. (1996). “Using small screen space more efficiently.”

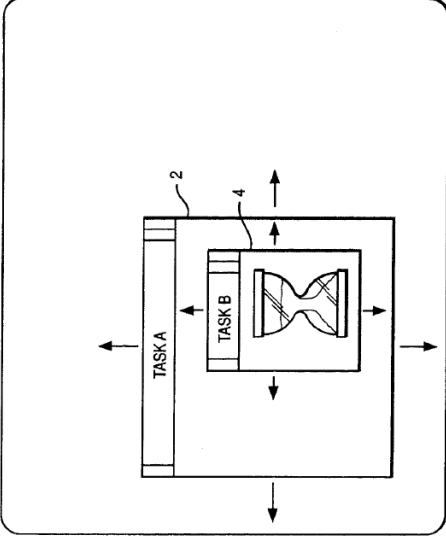
⁷ M. Lauff and W. Gellerson. Multimedia Client Implementation on Personal Digital Assistants. TecO.

⁸ Motorola continues to investigate other browsers that were in existence prior to the priority date of the ‘780 patent (e.g., NetHopper, Newt’s Cape, PocketWeb and Vagon).

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or various systems implementing a web browser on a mobile device window(s).” Col. 2, ll. 17-26
		<p>“The invention provides the advantage that display of the window(s) of the called task in a manner connected to the window(s) of the calling task, clarifies to the user the relationship between the tasks. Furthermore, the use of the data identifying the calling task’s window(s) makes it possible for the system to monitor any changes in the position or size of the calling task’s window(s) and make appropriate changes in size and position to the window(s) of the called task thereby preserving the visual relationship between the tasks.” Col. 2, ll. 27-36.</p> <p>“In preferred embodiments of the invention said second task’s window(s) is displayed within said first task’s window(s). This form of display provides an unmistakable visual link between the windows of the different tasks.” Col. 2, ll. 63-66.</p> <p>“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.</p> <p>“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 in a hypermedia browser having a content viewing area for viewing content. Hypermedia browsers download files from another system, and it would have therefore been obvious to one of skill in the art to implement Vaughton ‘744 in the context of a hypermedia browser.</p> <p>In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teach a hypermedia browser having a content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing area during times when the browser is loading content (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744</p>

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device	with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.2	wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area,	See Figure 2.	 <p style="text-align: center;">FIG. 2</p>
1.3	wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser.	See Figure 2, which illustrates the temporary graphic element as an hourglass. “It is a feature of advanced multitasking data processing systems that the tasks executed are becoming more interdependent . . . An example of such interdependence would be one task requiring a particular piece of information to be retrieved from a database by a database retrieval task.” Col. 1, ll. 59-65. “The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and	

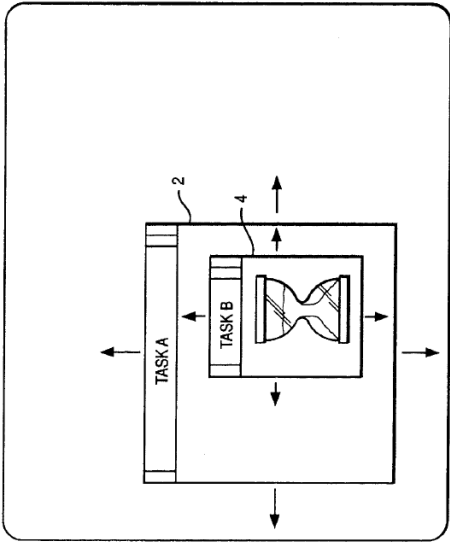
	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
		said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.
2	A hypermedia browser as recited in claim 1, wherein the browser is configured to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.	<p>“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.</p> <p>“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.</p>
3	A hypermedia browser as recited in claim 1, wherein the temporary graphic element indicates to a user that the browser is loading content.	“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.
4	A hypermedia browser as recited in claim 1, wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.	“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.
5	A hypermedia browser as recited in claim 1, wherein the temporary graphic element is animated.	“The window for Task B shows a sand glass in which the flow of the sand through the sand glass gives an indication of the progress of the downloading of the file in question.” Col. 3, ll. 37-40.
6	A hypermedia browser as recited in claim 1, wherein the hypermedia browser displays the temporary graphic element	“In preferred embodiments of the invention said second task’s window(s) is displayed within said first task’s window(s). This form of display provides an unmistakable visual link between the windows of the different tasks. It will however be appreciated

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or various systems implementing a web browser on a mobile device	that other ways of connecting the windows are possible, e.g. the window(s) of the second task may abut the edge of the window(s) of the first task--the windows may be “clipped together” at their edge.” Col. 2, l. 63 - Col. 3, l. 3. See Figure 2.
		 <p style="text-align: center;">FIG. 2</p>	
9	A hypermedia browser as recited in claim 1, wherein the temporary graphic element conveys status information of the browser.	“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.	
10	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language. Vaughton ‘744 discloses a	

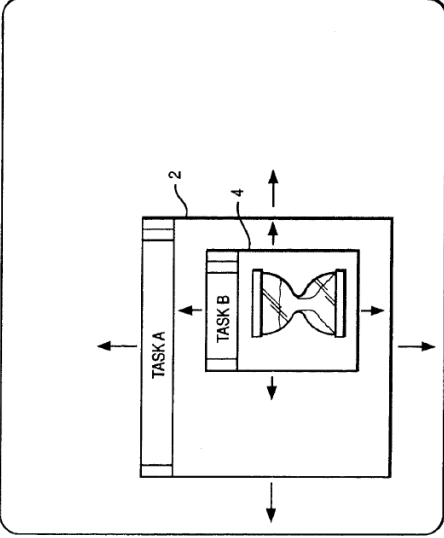
	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
		<p>system that has the function of downloading files from another system required by a first task. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Vaughton ‘744 in the context of a hypermedia browser.</p> <p>In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
11	<p>A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.</p>	<p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web. Vaughton ‘744 discloses a system that has the function of downloading files from another system required by a first task. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Vaughton ‘744 in the context of a hypermedia browser.</p> <p>In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643</p>

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or various systems implementing a web browser on a mobile device
		renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
12	12. An information processing device comprising:	“The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows[.]” Col. 2, ll. 17-20.
12.1	a processor;	“The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows[.]” Col. 2, ll. 17-20.
12.2	a display;	“The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows[.]” Col. 2, ll. 17-20.
12.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display;	The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows, wherein execution of a first task may trigger

	<p>U.S. Patent 6,339,780</p>	<p>U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device</p>
	<p>wherein the hypermedia browser has a content viewing area for viewing content and wherein the hypermedia browser displays a temporary graphic element over the content viewing area during times when the browser is loading visible content;</p>	<p>execution of a second task characterised in that when execution of said first task triggers execution of said second task, said processor uses data identifying said first task’s display window(s) and to control display of said second task’s display window(s) so as to be connected to and move with said first task’s display window(s). Col. 2, ll. 17-26</p> <p>The invention provides the advantage that display of the window(s) of the called task in a manner connected to the window(s) of the calling task, clarifies to the user the relationship between the tasks. Furthermore, the use of the data identifying the calling task’s window(s) makes it possible for the system to monitor any changes in the position or size of the calling task’s window(s) and make appropriate changes in size and position to the window(s) of the called task thereby preserving the visual relationship between the tasks. Col. 2, ll. 27-36.</p> <p>In preferred embodiments of the invention said second task’s window(s) is displayed within said first task’s window(s). This form of display provides an unmistakable visual link between the windows of the different tasks. Col. 2, ll. 63-66.</p> <p>“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.</p> <p>“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 in a hypermedia browser having a content viewing area for viewing content. Hypermedia browsers download files from another system, and it would have therefore been obvious to one of skill in the art to implement Vaughton ‘744 in the context of a hypermedia browser.</p> <p>In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teach a hypermedia</p>

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device	browser having a content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing area during times when the browser is loading visible content (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
12.4	wherein the temporary graphic element is positioned only over a portion of the content viewing area and obstructs only part of the visible content in the content viewing area; and	See Figure 2.	 <p>FIG. 2</p>
12.5	wherein the temporary graphic element indicates to a user that the browser is loading content and content comprises data for presentation which is from a	See Figure 2.	“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744” alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or various systems implementing a web browser on a mobile device
	source external to the browser.	said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8. “It is a feature of advanced multitasking data processing systems that the tasks executed are becoming more interdependent . . . An example of such interdependence would be one task requiring a particular piece of information to be retrieved from a database by a database retrieval task.” Col. 1, ll. 59-65. “The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.
13	An information processing device as recited in claim 12, wherein the temporary graphic element is animated.	The window for Task B shows a sand glass in which the flow of the sand through the sand glass gives an indication of the progress of the downloading of the file in question. Col. 3, ll. 37-40.
14	An information processing device as recited in claim 12, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	In preferred embodiments of the invention said second task’s window(s) is displayed within said first task’s window(s). This form of display provides an unmistakable visual link between the windows of the different tasks. It will however be appreciated that other ways of connecting the windows are possible, e.g. the window(s) of the second task may abut the edge of the window(s) of the first task--the windows may be “clipped together” at their edge. Col. 2, l. 63 - Col. 3, l. 3. See Figure 2.

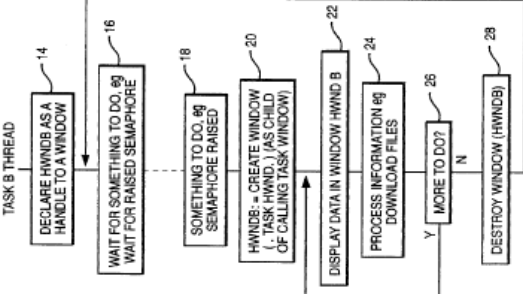
	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
17	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	 <p style="text-align: center;">FIG. 2</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language. Vaughton ‘744 discloses a system that has the function of downloading files from another system required by a first task. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Vaughton ‘744 in the context of a hypermedia browser.</p> <p>In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to</p>

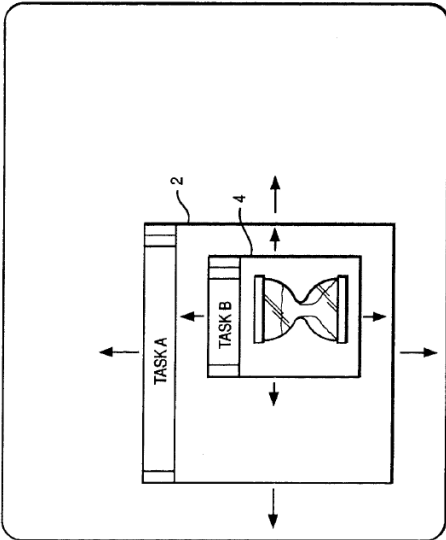
	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
		modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
18	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web. Vaughton ‘744 discloses a system that has the function of downloading files from another system required by a first task. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Vaughton ‘744 in the context of a hypermedia browser.</p> <p>In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of</p>

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
		ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
20	An information processing device as recited in claim 12, wherein the temporary graphic element is not content.	See Figure 2, which illustrates the temporary graphic element as an hourglass. “It is a feature of advanced multitasking data processing systems that the tasks executed are becoming more interdependent . . . An example of such interdependence would be one task requiring a particular piece of information to be retrieved from a database by a database retrieval task.” Col. 1, ll. 59-65. “The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.
21	An information processing device as recited in claim 12, wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.	“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.
32	32. A method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content, the method comprising:	See Figure 2. “The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 in a hypermedia browser having a content viewing area for viewing content. Hypermedia browsers download files from another system, and it would have therefore been obvious to one of skill in the art to

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device implement Vaughton ‘744 in the context of a hypermedia browser.
		In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teach a hypermedia browser having a content viewing area for viewing content (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
32.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen being without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;	<p>“The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows, wherein execution of a first task may trigger execution of a second task characterised in that when execution of said first task triggers execution of said second task, said processor uses data identifying said first task’s display window(s) and to control display of said second task’s display window(s) so as to be connected to and move with said first task’s display window(s).” Col. 2, ll. 17-26</p> <p>“The invention provides the advantage that display of the window(s) of the called task in a manner connected to the window(s) of the calling task, clarifies to the user the relationship between the tasks. Furthermore, the use of the data identifying the calling task’s window(s) makes it possible for the system to monitor any changes in the position or size of the calling task’s window(s) and make appropriate changes in size and position to the window(s) of the called task thereby preserving the visual relationship between the tasks.” Col. 2, ll. 27-36.</p> <p>“In preferred embodiments of the invention said second task’s window(s) is displayed within said first task’s window(s). This form of display provides an unmistakable</p>

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or various systems implementing a web browser on a mobile device
		visual link between the windows of the different tasks.” Col. 2, ll. 63-66. “The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8. “Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.
32.2	receiving an instruction to load new content into the content viewing area;	“It is a feature of advanced multitasking data processing systems that the tasks executed are becoming more interdependent, i.e. one task may start or call upon another task to carry out some function required by that first task. An example of such interdependence would be one task requiring a particular piece of information to be retrieved from a database by a database retrieval task. The calling task would start the database retrieval task and pass to it details concerning the data required.” Col. 1, ll. 59-67.
32.3	loading such new content into the content viewing area; and	See Fig. 3.

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or various systems implementing a web browser on a mobile device
		 <pre> graph TD 14[DECLARE HwndB AS A HANDLE TO A WINDOW] --> 16[WAIT FOR SOMETHING TO DO. #g WAIT FOR RAISED SEMAPHORE] 16 --> 18[SOMETHING TO DO. #g SEMAPHORE RAISED] 18 --> 20[HwndB -> CREATE WINDOW TASK Hwnd X (AS CHILD OF CALLING TASK WINDOW)] 20 --> 22[DISPLAY DATA IN WINDOW Hwnd B] 22 --> 24[PROCESS INFORMATION #g DOWNLOAD FILES] 24 --> 26{MORE TO DO?} 26 -- Y --> 24 26 -- N --> 28[DESTROY WINDOW (HwndB)] </pre>
32.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	<p>“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.</p> <p>See Fig. 2</p>

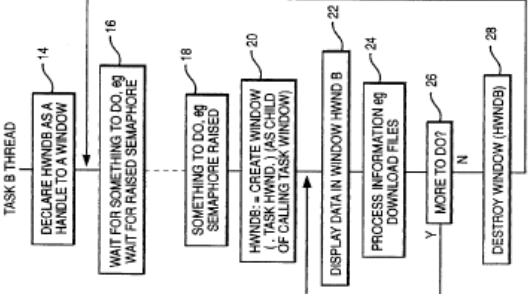
	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or various systems implementing a web browser on a mobile device
		 <p style="text-align: center;">FIG. 2</p> <p>“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.</p>
32.5	wherein content comprises data for presentation which is from a source external to the browser.	<p>“It is a feature of advanced multitasking data processing systems that the tasks executed are becoming more interdependent . . . An example of such interdependence would be one task requiring a particular piece of information to be retrieved from a database by a database retrieval task.” Col. 1, ll. 59-65.</p> <p>“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.</p>
33	A method as recited in claim 32 further comprising, upon completion of the loading, removing the “load status”	“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task

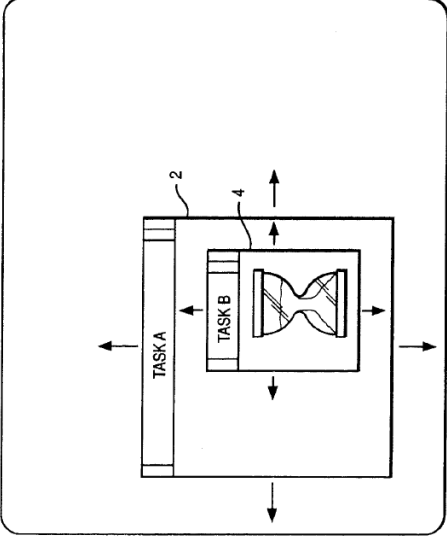
	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
	graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.
34	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language. Vaughton ‘744 discloses a system that has the function of downloading files from another system required by a first task. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Vaughton ‘744 in the context of a hypermedia browser.</p> <p>In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
35	A hypermedia browser of claim 32,	One of ordinary skill in the art would recognize that a hypermedia browser has the

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
	wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web.	function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web. Vaughton ‘744 discloses a system that has the function of downloading files from another system required by a first task. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Vaughton ‘744 in the context of a hypermedia browser. In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
36	A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content,	The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows, wherein execution of a first task may trigger execution of a second task characterised in that when execution of said first task triggers execution of said second task, said processor uses data identifying said first task’s display window(s) and to control display of said second task’s display

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or various systems implementing a web browser on a mobile device
	the method comprising:	<p>window(s) so as to be connected to and move with said first task’s display window(s). Col. 2, ll. 17-26</p> <p>The invention provides the advantage that display of the window(s) of the called task in a manner connected to the window(s) of the calling task, clarifies to the user the relationship between the tasks. Furthermore, the use of the data identifying the calling task’s window(s) makes it possible for the system to monitor any changes in the position or size of the calling task’s window(s) and make appropriate changes in size and position to the window(s) of the called task thereby preserving the visual relationship between the tasks. Col. 2, ll. 27-36.</p> <p>In preferred embodiments of the invention said second task’s window(s) is displayed within said first task’s window(s). This form of display provides an unmistakable visual link between the windows of the different tasks. Col. 2, ll. 63-66.</p> <p>“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 in a hypermedia browser having a content viewing area for viewing content. Hypermedia browsers download files from another system, and it would have therefore been obvious to one of skill in the art to implement Vaughton ‘744 in the context of a hypermedia browser.</p> <p>In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teach a hypermedia browser having a content viewing area for viewing content (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period.</p>

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
		Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
36.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen is without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;	<p>The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows, wherein execution of a first task may trigger execution of a second task characterised in that when execution of said first task triggers execution of said second task, said processor uses data identifying said first task’s display window(s) and to control display of said second task’s display window(s) so as to be connected to and move with said first task’s display window(s). Col. 2, ll. 17-26</p> <p>The invention provides the advantage that display of the window(s) of the called task in a manner connected to the window(s) of the calling task, clarifies to the user the relationship between the tasks. Furthermore, the use of the data identifying the calling task’s window(s) makes it possible for the system to monitor any changes in the position or size of the calling task’s window(s) and make appropriate changes in size and position to the window(s) of the called task thereby preserving the visual relationship between the tasks. Col. 2, ll. 27-36.</p> <p>In preferred embodiments of the invention said second task’s window(s) is displayed within said first task’s window(s). This form of display provides an unmistakable visual link between the windows of the different tasks. Col. 2, ll. 63-66.</p> <p>“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.</p> <p>“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.</p>

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
36.2	receiving an instruction to load new content into the content viewing area;	“It is a feature of advanced multitasking data processing systems that the tasks executed are becoming more interdependent, i.e. one task may start or call upon another task to carry out some function required by that first task. An example of such interdependence would be one task requiring a particular piece of information to be retrieved from a database by a database retrieval task. The calling task would start the database retrieval task and pass to it details concerning the data required.” Col. 1, ll. 59-67.
36.3	loading such new content into the content viewing area; and	See Fig. 3. 
36.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8. See Fig. 2

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or various systems implementing a web browser on a mobile device
		 <p style="text-align: center;">FIG. 2</p> <p>“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.</p>
36.5	wherein content comprises data for presentation which is from a source external to the browser.	<p>“It is a feature of advanced multitasking data processing systems that the tasks executed are becoming more interdependent . . . An example of such interdependence would be one task requiring a particular piece of information to be retrieved from a database by a database retrieval task.” Col. 1, ll. 59-65.</p> <p>“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.</p>
37	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a	One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 with a hypermedia

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
	markup language, visible text of such a markup language, and visible results of a scripting language.	browser wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language. Vaughton ‘744 discloses a system that has the function of downloading files from another system required by a first task. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Vaughton ‘744 in the context of a hypermedia browser. In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
38	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web. Vaughton ‘744 discloses a system that has the function of downloading files from another system required by a first task. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Vaughton ‘744 in the context of a

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device and/or hypermedia browser.
		In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
39	A computer-readable medium as recited in claim 36 further having additional computer-executable instructions that perform a method comprising, upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.
40	40. An information processing device comprising:	“The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows[.]” Col. 2, ll. 17-20.

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
40.1	a processor;	“The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows[.]” Col. 2, ll. 17-20.
40.2	a display;	“The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows[.]” Col. 2, ll. 17-20.
40.2	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser is configured to operate in a content-loading mode and a content-loaded mode;	<p>The invention provides a data processing system having a window based display and a processor executing a plurality of interleaved data processing tasks each having one or more associated display windows, wherein execution of a first task may trigger execution of a second task characterised in that when execution of said first task triggers execution of said second task, said processor uses data identifying said first task’s display window(s) and to control display of said second task’s display window(s) so as to be connected to and move with said first task’s display window(s). Col. 2, ll. 17-26</p> <p>The invention provides the advantage that display of the window(s) of the called task in a manner connected to the window(s) of the calling task, clarifies to the user the relationship between the tasks. Furthermore, the use of the data identifying the calling task’s window(s) makes it possible for the system to monitor any changes in the position or size of the calling task’s window(s) and make appropriate changes in size and position to the window(s) of the called task thereby preserving the visual relationship between the tasks. Col. 2, ll. 27-36.</p> <p>In preferred embodiments of the invention said second task’s window(s) is displayed within said first task’s window(s). This form of display provides an unmistakable visual link between the windows of the different tasks. Col. 2, ll. 63-66.</p> <p>“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.</p> <p>“Task B puts up the window (4) for Task B as a child window of the window (2) of</p>

	U.S. Patent 6,339,780	<p>U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device</p> <p>Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 in a hypermedia browser having a content viewing area for viewing content. Hypermedia browsers download files from another system, and it would have therefore been obvious to one of skill in the art to implement Vaughton ‘744 in the context of a hypermedia browser.</p> <p>In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teach a hypermedia browser executing on the processor to load and display content in a content viewing area on the display (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
40.3	<p>in the content-loaded mode, the hypermedia browser displays loaded content in the content viewing area and no “load status” graphic element is displayed, wherein absence of such “load status” graphic element indicates that the browser is in the content-loaded mode;</p>	<p>“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.</p>
40.4	<p>in the content-loading mode, the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a</p>	<p>“Task B puts up the window (4) for Task B as a child window of the window (2) of Task A . . . Task B destroys the window (4) for Task B, when the processing for Task B is completed.” Col. 3, ll. 66-67 to Col. 4, ll. 1-2.</p>

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
	“load status” graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such “load status” graphic element indicates that the browser is in the content-loading mode; and	
40.5	wherein content comprises data for presentation which is from a source external to the browser.	<p>“It is a feature of advanced multitasking data processing systems that the tasks executed are becoming more interdependent . . . An example of such interdependence would be one task requiring a particular piece of information to be retrieved from a database by a database retrieval task.” Col. 1, ll. 59-65.</p> <p>“The invention has been found to be particularly advantageous when said system has the function of downloading files from another system required by said first task and said second task’s window(s) provides an indication of the progress of said downloading.” Col. 3, ll. 4-8.</p>
41	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a scripting language, and visible results of a	<p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language. Vaughton ‘744 discloses a system that has the function of downloading files from another system required by a first task. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Vaughton ‘744 in the context of a hypermedia browser.</p> <p>In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a</p>

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
		group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
42	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web.	<p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Vaughton ‘744 with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web. Vaughton ‘744 discloses a system that has the function of downloading files from another system required by a first task. Based on this, one of ordinary skill in the art would have been motivated to implement the teachings of Vaughton ‘744 in the context of a hypermedia browser.</p> <p>In addition, the combination of Vaughton ‘744 and Nguyen ‘498 and/or Judson ‘643 renders this claim obvious. Nguyen ‘498 and/or Judson ‘643 teaches a hypermedia browser, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web (see analysis of Nguyen ‘498 and/or Judson ‘643 in claim chart) and it would be obvious to one of skill in the art to modify Vaughton ‘744 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the</p>

	U.S. Patent 6,339,780	U.S. Patent 5,528,744 (“Vaughton ‘744’”) alone and/or in combination with U.S. Patent 6,584,498 (“Nguyen ‘498’”) and/or U.S. Patent 5,572,643 (“Judson ‘643’”) and/or various systems implementing a web browser on a mobile device
		same field and share common subject matter (systems for downloading files from another system), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

EXHIBIT B

Exemplar Chart of U.S. Patent 6,339,780**U.S. Patent 5,572,643 (“Judson ‘643’”)¹
Claims 1-6, 9-14, 17, 18, 20, 21 and 32-42**

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643’”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744’”) and/or various systems implementing a web browser on a mobile device
1	A hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area,	<p>“A method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser. The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server.” Abstract.</p> <p>“According to the preferred embodiment, there is described a method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser.” Col. 2, ll. 29-32.</p> <p>“Each of the clients may run a ‘browser,’ which is a known software tool used to access the servers via the access providers.” Col. 3, ll. 51-53.</p> <p>“In addition, although the various methods described are conveniently implemented in a general purpose computer selectively activated or reconfigured by software, one of ordinary skill in the art would also recognize that such methods may be carried out in hardware, in firmware, or in more specialized apparatus constructed to perform the required method steps.” Col. 7, ll. 52-59.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one</p>

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Web Browser with Dynamic Display of Information Objects During Linking” issued to Judson; filed October 19, 1995; issued November 5, 1996.

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>of skill in the art to implement a browser on an information processing device having a limited display area. Judson ‘643 discloses that “one of ordinary skill in the art would also recognize that [the invention] may be carried out in hardware, in firmware, or in more specialized apparatus constructed to perform the required method steps,” e.g., in a device having a limited display area. Based on this, one of skill in the art would have been motivated to implement a browser on an information processing device having a limited display area.</p> <p>In addition, the combination of Judson ‘643 and one or more references describing handheld devices with browsers renders this claim obvious. Gessler², Cooper³, Bartlett⁴, Watson⁵, Kamba⁶, and Lauff⁷ each teach a hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area (e.g., a PDA, handheld, mobile device, etc.), and it would be obvious to one of skill in the art to modify Judson ‘643 with these teachings.⁸ One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for</p>

² S. Gessler and A. Kotulla. PDAs as Mobile WWW Browsers. Computer Networks and ISDN Systems 28 (1995) 53-59 (“Gessler”) (see pgs. 53 and 56-58).

³ I. Cooper and R. Shufflebotham. PDWeb Browsers: Implementation Issues. (University of Kent at Canterbury, November 9, 1995) (“Cooper”) (see pgs. 3 and 5).

⁴ J. Bartlett. Experience with a Wireless World Wide Web Client. IEEE COMPCON 95 (San Francisco, March 1995) (“Bartlett”) (see pgs. 1, 4 and 5).

⁵ T. Watson. “Application Design for Wireless Computing” (1994 Mobile Computing Systems and Applications Workshop Position Paper (August 1994)).

⁶ T. Kamba, S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. (1996). “Using small screen space more efficiently.”

⁷ M. Lauff and W. Gellerson. Multimedia Client Implementation on Personal Digital Assistants. TecO.

⁸ Motorola continues to investigate other browsers that were in existence prior to the priority date of the ‘780 patent (e.g., NetHopper, Newt’s Cape, PocketWeb and Vagon).

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>downloading files from another system). Additionally, Bartlett cites Gessler and Cooper cites Gessler and Bartlett. Moreover, all of these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.1	<p>wherein the hypermedia browser has a content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing area during times when the browser is loading content,</p>	<p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“It is thus a primary object of the invention to enhance the operation of a web browser by causing the display of some useful information to the user during the period of user ‘downtime’ that otherwise occurs between linking and downloading of a hypertext document identified by the link.” Col. 1, ll. 59-63.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p>
1.2	<p>wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area,</p>	<p>“FIG. 8 is a representative screen display illustrating how the information object appears as a ‘mini’ web page upon activation of the hypertext link in the web page of FIG. 5.” Col. 3, ll. 38-40.</p> <p>“As used herein, the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above.” Col. 7, ll. 39-44.</p> <p>“Thus, an ‘information object’ according to the invention may include an applet which, for example, may generate an animated figure or icon, some aural output, a scrolling display, or a combination thereof.” Col. 8, ll. 9-12.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to position the temporary graphic element over the content viewing</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>area to obstruct only part of the content in the content viewing area. Judson ‘643 discloses that “the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above,” e.g., a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area.</p> <p>In addition, the combination of Judson ‘643 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Judson ‘643 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.3	wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser.	<p>“The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server. In response to the user clicking on the link, the link is activated by the browser to thereby request downloading of the hypertext document from the remote server to the graphical user interface of the client.” Col. 2, ll. 32-39.</p> <p>“Or, the information object(s) may have no direct relation to the content of any link in the document.” Col. 5, ll. 28-30.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>“The information object need not be embedded within an existing web page, but rather may be embedded within the home page of the browser or supported elsewhere within the client itself.” Col. 6, l. 66-Col. 7, l. 2.</p>
2	A hypermedia browser as recited in claim 1, wherein the browser is configured to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.	<p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p>
3	A hypermedia browser as recited in claim 1, wherein the temporary graphic element indicates to a user that the browser is loading content.	<p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to use the temporary graphic element to indicate to a user that the browser is loading content. For example, Judson ‘643 states “As used herein, the “information object” or “information” output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above.” Col. 7, ll. 38-44). Moreover, Vaughton ‘744 indicates, e.g., that “[t]he window for Task B shows a sand glass in which the flow of the sand through the sand glass gives an indication of the progress of the downloading of the file in question.” Vaughton ‘744 at Col. 3, ll. 37-40. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
4	A hypermedia browser as recited in claim 1, wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.	<p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to have the temporary graphic element disappear when the browser’s loading of content is complete to indicate to a user that such loading of content is complete. Moreover, Vaughton ‘744 indicates, e.g., that “Task B destroys the window (4) for Task B, when the processing for Task B is complete.” Vaughton ‘744 at Col. 4, ll. 1-2. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>“Thus, an ‘information object’ according to the invention may include an applet which, for example, may generate an animated figure or icon, some aural output, a scrolling display, or a combination thereof.” Col. 8, ll. 9-12.</p>
5	A hypermedia browser as recited in claim 1, wherein the temporary graphic element is animated.	
6	A hypermedia browser as recited in claim 1, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area. Judson ‘643 discloses that “[w]hile the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Based on this, one of skill in the art would have been

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>motivated to implement a browser wherein the one or more different types of informational messages to the user is displayed in a corner of the content viewing area. Additionally, displaying the one or more different types of informational messages to the user in a corner of the content viewing area is an obvious design choice to minimize the amount of content obstructed and would have been obvious to try.</p> <p>Moreover, Vaughton ‘744 indicates, e.g., that “the window(s) of the second task may abut the edge of the window(s) of the first task--the windows may be “clipped together” at their edge.” Vaughton ‘744 at Col. 2, l. 63 - Col. 3, l. 3. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
9	A hypermedia browser as recited in claim 1, wherein the temporary graphic element conveys status information of the browser.	<p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to use the temporary graphic element to convey status information of the browser. For example, Judson ‘643 states “As used herein, the “information object” or “information” output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above.” Col. 7, ll. 38-44). Moreover, Vaughton ‘744 indicates, e.g., that “[t]he window for Task B shows a sand glass in which the flow of the sand through the sand</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		glass gives an indication of the progress of the downloading of the file in question.” Vaughton ‘744 at Col. 3, ll. 37-40. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
10	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The ‘World Wide Web’ (WWW) is that collection of servers of the Internet that utilize the Hypertext Transfer Protocol (HTTP). HTTP is a known application protocol that provides users access to files (which can be in different formats such as text, graphics, images, sound, video, etc.) using a standard page description language known as Hypertext Markup Language (HTML). HTML provides basic document formatting and allows the developer to specify “links” to other servers and files. Use of an HTML-compliant client browser involves specification of a link via the URL. Upon such specification, the client makes a tcp/ip request to the server identified in the link and receives a ‘web page’ (namely, a document formatted according to HTML) in return.” Col. 3, l. 58-Col. 4, l. 4.</p> <p>“RAM 34 also supports a number of Internet access tools including, for example, the HTTP-compliant web browser 62. Known browser software includes Netscape, Netscape Navigator 2.0, Mosaic, and the like. The present invention is designed to operate within any of these known or developing web browsers, which are preferably modified as described herein to achieve the dynamic display of information objects during web site linking activities.” Col. 4, ll. 39-47.</p> <p>“As noted above, a web browser 62 running on the client uses a TCP/IP connection to pass a request to a web server running a HTTP “service” (under the WINDOWS operating system) or “daemon” (under the UNIX operating system). The HTTP service then responds to the request, typically by sending a “web page” formatted in the Hypertext Markup Language, or HTML, to the browser. The browser then displays the web page using local resources (e.g., fonts and colors).” Col. 5, ll. 41-49.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
11	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>“The ‘World Wide Web’ (WWW) is that collection of servers of the Internet that utilize the Hypertext Transfer Protocol (HTTP). HTTP is a known application protocol that provides users access to files (which can be in different formats such as text, graphics, images, sound, video, etc.) using a standard page description language known as Hypertext Markup Language (HTML). HTML provides basic document formatting and allows the developer to specify “links” to other servers and files. Use of an HTML-compliant client browser involves specification of a link via the URL. Upon such specification, the client makes a tcp/ip request to the server identified in the link and receives a ‘web page’ (namely, a document formatted according to HTML) in return.” Col. 3, l. 58-Col. 4, l. 4.</p> <p>“RAM 34 also supports a number of Internet access tools including, for example, the HTTP-compliant web browser 62. Known browser software includes Netscape, Netscape Navigator 2.0, Mosaic, and the like. The present invention is designed to operate within any of these known or developing web browsers, which are preferably modified as described herein to achieve the dynamic display of information objects during web site linking activities.” Col. 4, ll. 39-47.</p> <p>“As noted above, a web browser 62 running on the client uses a TCP/IP connection to pass a request to a web server running a HTTP “service” (under the WINDOWS operating system) or “daemon” (under the UNIX operating system). The HTTP service then responds to the request, typically by sending a “web page” formatted in the Hypertext Markup Language, or HTML, to the browser. The browser then displays the web page using local resources (e.g., fonts and colors).” Col. 5, ll. 41-49.</p> <p>“A method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser.” Abstract.</p> <p>“The microprocessor 32 is connected to the system bus 31 and is supported by read only memory (ROM) 33 and random access memory (RAM) 34 also connected to system bus 31.” Col. 4, ll. 10-13.</p> <p>“The keyboard controller 38 provides the hardware interface for the keyboard 22, the controller 39 provides the hardware interface for the mouse (or other point and click</p>
12	An information processing device comprising:	
12.1	a processor;	
12.2	a display;	

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		device) 23, the video controller 40 is the hardware interface for the display 24, and the audio controller 41 is the hardware interface for the multimedia speakers 25a and 25b.” Col. 4, ll. 27-33.
12.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser has a content viewing area for viewing content and wherein the hypermedia browser displays a temporary graphic element over the content viewing area during times when the browser is loading visible content;	<p>“A method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser. The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server.” Abstract.</p> <p>“According to the preferred embodiment, there is described a method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser.” Col. 2, ll. 29-32.</p> <p>“Each of the clients may run a ‘browser,’ which is a known software tool used to access the servers via the access providers.” Col. 3, ll. 51-53.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to use the temporary graphic element to indicate to a user that the browser is loading content. For example, Judson ‘643 states “As used herein, the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above.” Col. 7, ll. 38-44). Moreover, Vaughton ‘744 indicates, e.g., that “[t]he window for Task B shows a sand glass in which the flow of the sand through the sand glass gives an indication of the progress of the downloading of the file in question.” Vaughton ‘744 at Col. 3, ll. 37-40. One of skill in the art would be</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
12.4	wherein the temporary graphic element is positioned only over a portion of the content viewing area and obstructs only part of the visible content in the content viewing area; and	<p>“FIG. 8 is a representative screen display illustrating how the information object appears as a ‘mini’ web page upon activation of the hypertext link in the web page of FIG. 5.” Col. 3, ll. 38-40.</p> <p>“As used herein, the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above.” Col. 7, ll. 39-44.</p> <p>“Thus, an ‘information object’ according to the invention may include an applet which, for example, may generate an animated figure or icon, some aural output, a scrolling display, or a combination thereof.” Col. 8, ll. 9-12.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to position the temporary graphic element over the content viewing area to obstruct only part of the content in the content viewing area. Judson ‘643 discloses that “the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above,” e.g., a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area.</p> <p>In addition, the combination of Judson ‘643 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element positioned over the</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>content viewing area to obstruct only part of the content in the content viewing area (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Judson ‘643 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
12.5	<p>wherein the temporary graphic element indicates to a user that the browser is loading content and content comprises data for presentation which is from a source external to the browser.</p>	<p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server. In response to the user clicking on the link, the link is activated by the browser to thereby request downloading of the hypertext document from the remote server to the graphical user interface of the client.” Col. 2, ll. 32-39.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to use the temporary graphic element to indicate to a user that the browser is loading content. For example, Judson ‘643 states “As used herein, the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above.” Col. 7, ll. 38-44). Moreover, Vaughton ‘744 indicates, e.g., that</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>“[t]he window for Task B shows a sand glass in which the flow of the sand through the sand glass gives an indication of the progress of the downloading of the file in question.” Vaughton ‘744 at Col. 3, ll. 37-40. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
13	An information processing device as recited in claim 12, wherein the temporary graphic element is animated.	<p>“Thus, an ‘information object’ according to the invention may include an applet which, for example, may generate an animated figure or icon, some aural output, a scrolling display, or a combination thereof.” Col. 8, ll. 9-12.</p>
14	An information processing device as recited in claim 12, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area. Judson ‘643 discloses that “[w]hile the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Based on this, one of skill in the art would have been motivated to implement a browser wherein the one or more different types of informational messages to the user is displayed in a corner of the content viewing area. Additionally, displaying the one or more different types of informational messages to the user in a corner of the content viewing area is an obvious design choice to minimize the amount of content obstructed and would have been obvious to try.</p> <p>Moreover, Vaughton ‘744 indicates, e.g., that “the window(s) of the second task may abut the edge of the window(s) of the first task--the windows may be “clipped together” at their edge.” Vaughton ‘744 at Col. 2, l. 63 - Col. 3, l. 3. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
17	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The ‘World Wide Web’ (WWW) is that collection of servers of the Internet that utilize the Hypertext Transfer Protocol (HTTP). HTTP is a known application protocol that provides users access to files (which can be in different formats such as text, graphics, images, sound, video, etc.) using a standard page description language known as Hypertext Markup Language (HTML). HTML provides basic document formatting and allows the developer to specify “links” to other servers and files. Use of an HTML-compliant client browser involves specification of a link via the URL. Upon such specification, the client makes a tcp/ip request to the server identified in the link and receives a ‘web page’ (namely, a document formatted according to HTML) in return.” Col. 3, l. 58-Col. 4, l. 4.</p> <p>“RAM 34 also supports a number of Internet access tools including, for example, the HTTP-compliant web browser 62. Known browser software includes Netscape, Netscape Navigator 2.0, Mosaic, and the like. The present invention is designed to operate within any of these known or developing web browsers, which are preferably modified as described herein to achieve the dynamic display of information objects during web site linking activities.” Col. 4, ll. 39-47.</p> <p>“As noted above, a web browser 62 running on the client uses a TCP/IP connection to pass a request to a web server running a HTTP “service” (under the WINDOWS operating system) or “daemon” (under the UNIX operating system). The HTTP service then responds to the request, typically by sending a “web page” formatted in the Hypertext Markup Language, or HTML, to the browser. The browser then displays the web page using local resources (e.g., fonts and colors).” Col. 5, ll. 41-49.</p>
18	A hypermedia browser of claim 12, wherein content is data formatted for	“The ‘World Wide Web’ (WWW) is that collection of servers of the Internet that utilize the Hypertext Transfer Protocol (HTTP). HTTP is a known application

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
	presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>protocol that provides users access to files (which can be in different formats such as text, graphics, images, sound, video, etc.) using a standard page description language known as Hypertext Markup Language (HTML). HTML provides basic document formatting and allows the developer to specify “links” to other servers and files. Use of an HTML-compliant client browser involves specification of a link via the URL. Upon such specification, the client makes a tcp/ip request to the server identified in the link and receives a ‘web page’ (namely, a document formatted according to HTML) in return.” Col. 3, l. 58-Col. 4, l. 4.</p> <p>“RAM 34 also supports a number of Internet access tools including, for example, the HTTP-compliant web browser 62. Known browser software includes Netscape, Netscape Navigator 2.0, Mosaic, and the like. The present invention is designed to operate within any of these known or developing web browsers, which are preferably modified as described herein to achieve the dynamic display of information objects during web site linking activities.” Col. 4, ll. 39-47.</p> <p>“As noted above, a web browser 62 running on the client uses a TCP/IP connection to pass a request to a web server running a HTTP “service” (under the WINDOWS operating system) or “daemon” (under the UNIX operating system). The HTTP service then responds to the request, typically by sending a “web page” formatted in the Hypertext Markup Language, or HTML, to the browser. The browser then displays the web page using local resources (e.g., fonts and colors).” Col. 5, ll. 41-49.</p>
20	An information processing device as recited in claim 12, wherein the temporary graphic element is not content.	<p>“Or, the information object(s) may have no direct relation to the content of any link in the document.” Col. 5, ll. 28-30.</p> <p>“The information object need not be embedded within an existing web page, but rather may be embedded within the home page of the browser or supported elsewhere within the client itself.” Col. 6, l. 66-Col. 7, l. 2.</p>
21	An information processing device as recited in claim 12, wherein the temporary graphic element disappears when the browser’s loading of content is	<p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
	complete to indicate to a user that such loading of content is complete.	downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to have the temporary graphic element disappear when the browser’s loading of content is complete to indicate to a user that such loading of content is complete. Moreover, Vaughton ‘744 indicates, e.g., that “Task B destroys the window (4) for Task B, when the processing for Task B is complete.” Vaughton ‘744 at Col. 4, ll. 1-2. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
32	A method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content, the method comprising:	<p>“A method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser. The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server.” Abstract.</p> <p>“According to the preferred embodiment, there is described a method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser.” Col. 2, ll. 29-32.</p> <p>“Each of the clients may run a ‘browser,’ which is a known software tool used to access the servers via the access providers.” Col. 3, ll. 51-53.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device messages to the user.” Col. 2, ll. 39-42. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to use the temporary graphic element to convey status information of the browser. For example, Judson ‘643 states “As used herein, the “information object” or “information” output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above.” Col. 7, ll. 38-44). Moreover, Vaughton ‘744 indicates, e.g., that “[t]he window for Task B shows a sand glass in which the flow of the sand through the sand glass gives an indication of the progress of the downloading of the file in question.” Vaughton ‘744 at Col. 3, ll. 37-40. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
32.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen being without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;	“The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server.” Col. 2, ll. 32-35. “While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract. “While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser, “the screen being without a “load status” graphic element.” Judson ‘643 discloses that “one of ordinary skill in the art would

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		also recognize that [the invention] may be carried out in hardware, in firmware, or in more specialized apparatus constructed to perform the required method steps,” e.g., to save space on a display with a limited display area. Based on this, one of skill in the art would have been motivated to implement a browser having a screen without a “load status” graphic element. Additionally, having a screen without a “load status” graphic element is an obvious design choice where display area is limited and would have been obvious to try. See claim 1 above.
32.2	receiving an instruction to load new content into the content viewing area;	<p>“In response to the user clicking on the link, the link is activated by the browser to thereby request downloading of the hypertext document from the remote server to the graphical user interface of the client.” Col. 2, ll. 35-39.</p> <p>“In response to the user pointing and clicking on the link, the link is said to be ‘activated’ to begin the download of the linked document or text.” Col. 4, ll. 62-64.</p>
32.3	loading such new content into the content viewing area; and	“In response to the user clicking on the link, the link is activated by the browser to thereby request downloading of the hypertext document from the remote server to the graphical user interface of the client. While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 35-42.
32.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	<p>“FIG. 8 is a representative screen display illustrating how the information object appears as a ‘mini’ web page upon activation of the hypertext link in the web page of FIG. 5.” Col. 3, ll. 38-40.</p> <p>“As used herein, the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above.” Col. 7, ll. 39-44.</p> <p>“Thus, an ‘information object’ according to the invention may include an applet which, for example, may generate an animated figure or icon, some aural output, a scrolling display, or a combination thereof.” Col. 8, ll. 9-12.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 ("Judson '643") alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744") and/or various systems implementing a web browser on a mobile device
		<p>of skill in the art to position the temporary graphic element over the content viewing area to obstruct only part of the content in the content viewing area. Judson '643 discloses that "the 'information object' or 'information' output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above," e.g., a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area.</p> <p>In addition, the combination of Judson '643 and Vaughton '744 renders this claim obvious. Vaughton '744 teaches a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area (see analysis of Vaughton '744 in claim chart) and it would be obvious to one of skill in the art to modify Judson '643 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
32.5	wherein content comprises data for presentation which is from a source external to the browser.	"The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server. In response to the user clicking on the link, the link is activated by the browser to thereby request downloading of the hypertext document from the remote server to the graphical user interface of the client." Col. 2, ll. 32-39.
33	A method as recited in claim 32 further	"While the client waits for a reply and/or as the hypertext document is being

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
	comprising, upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	<p>downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to upon completion of the loading, remove the “load status” graphic element. Moreover, Vaughton ‘744 indicates, e.g., that “Task B destroys the window (4) for Task B, when the processing for Task B is complete.” Vaughton ‘744 at Col. 4, ll. 1-2. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
34	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The ‘World Wide Web’ (WWW) is that collection of servers of the Internet that utilize the Hypertext Transfer Protocol (HTTP). HTTP is a known application protocol that provides users access to files (which can be in different formats such as text, graphics, images, sound, video, etc.) using a standard page description language known as Hypertext Markup Language (HTML). HTML provides basic document formatting and allows the developer to specify “links” to other servers and files. Use of an HTML-compliant client browser involves specification of a link via the URL. Upon such specification, the client makes a tcp/ip request to the server identified in the link and receives a ‘web page’ (namely, a document formatted according to HTML) in return.” Col. 3, l. 58-Col. 4, l. 4.</p> <p>“RAM 34 also supports a number of Internet access tools including, for example, the HTTP-compliant web browser 62. Known browser software includes Netscape, Netscape Navigator 2.0, Mosaic, and the like. The present invention is designed to operate within any of these known or developing web browsers, which are preferably</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		modified as described herein to achieve the dynamic display of information objects during web site linking activities.” Col. 4, ll. 39-47. “As noted above, a web browser 62 running on the client uses a TCP/IP connection to pass a request to a web server running a HTTP “service” (under the WINDOWS operating system) or “daemon” (under the UNIX operating system). The HTTP service then responds to the request, typically by sending a “web page” formatted in the Hypertext Markup Language, or HTML, to the browser. The browser then displays the web page using local resources (e.g., fonts and colors).” Col. 5, ll. 41-49.
35	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>“The ‘World Wide Web’ (WWW) is that collection of servers of the Internet that utilize the Hypertext Transfer Protocol (HTTP). HTTP is a known application protocol that provides users access to files (which can be in different formats such as text, graphics, images, sound, video, etc.) using a standard page description language known as Hypertext Markup Language (HTML). HTML provides basic document formatting and allows the developer to specify “links” to other servers and files. Use of an HTML-compliant client browser involves specification of a link via the URL. Upon such specification, the client makes a tcp/ip request to the server identified in the link and receives a ‘web page’ (namely, a document formatted according to HTML) in return.” Col. 3, l. 58-Col. 4, l. 4.</p> <p>“RAM 34 also supports a number of Internet access tools including, for example, the HTTP-compliant web browser 62. Known browser software includes Netscape, Netscape Navigator 2.0, Mosaic, and the like. The present invention is designed to operate within any of these known or developing web browsers, which are preferably modified as described herein to achieve the dynamic display of information objects during web site linking activities.” Col. 4, ll. 39-47.</p> <p>“As noted above, a web browser 62 running on the client uses a TCP/IP connection to pass a request to a web server running a HTTP “service” (under the WINDOWS operating system) or “daemon” (under the UNIX operating system). The HTTP service then responds to the request, typically by sending a “web page” formatted in the Hypertext Markup Language, or HTML, to the browser. The browser then</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device displays the web page using local resources (e.g., fonts and colors).” Col. 5, ll. 41-49.
36	A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content, the method comprising:	<p>“A method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser. The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server.” Abstract.</p> <p>“According to the preferred embodiment, there is described a method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser.” Col. 2, ll. 29-32.</p> <p>“Each of the clients may run a ‘browser,’ which is a known software tool used to access the servers via the access providers.” Col. 3, ll. 51-53.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to use the temporary graphic element to convey status information of the browser. For example, Judson ‘643 states “As used herein, the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above.” Col. 7, ll. 38-44). Moreover, Vaughton ‘744 indicates, e.g., that “[t]he window for Task B shows a sand glass in which the flow of the sand through the sand glass gives an indication of the progress of the downloading of the file in question.” Vaughton ‘744 at Col. 3, ll. 37-40. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
36.1	<p>displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen is without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;</p>	<p>“The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server.” Col. 2, ll. 32-35.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser, “the screen being without a “load status” graphic element.” Judson ‘643 discloses that “one of ordinary skill in the art would also recognize that [the invention] may be carried out in hardware, in firmware, or in more specialized apparatus constructed to perform the required method steps,” e.g., to save space on a display with a limited display area. Based on this, one of skill in the art would have been motivated to implement a browser having a screen without a “load status” graphic element. Additionally, having a screen without a “load status” graphic element is an obvious design choice where display area is limited and would have been obvious to try. See claim 1 above.</p>
36.2	<p>receiving an instruction to load new content into the content viewing area;</p>	<p>“In response to the user clicking on the link, the link is activated by the browser to thereby request downloading of the hypertext document from the remote server to the graphical user interface of the client.” Col. 2, ll. 35-39.</p> <p>“In response to the user pointing and clicking on the link, the link is said to be ‘activated’ to begin the download of the linked document or text.” Col. 4, ll. 62-64.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
36.3	loading such new content into the content viewing area; and	“In response to the user clicking on the link, the link is activated by the browser to thereby request downloading of the hypertext document from the remote server to the graphical user interface of the client. While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 35-42.
36.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	<p>“FIG. 8 is a representative screen display illustrating how the information object appears as a ‘mini’ web page upon activation of the hypertext link in the web page of FIG. 5.” Col. 3, ll. 38-40.</p> <p>“As used herein, the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above.” Col. 7, ll. 39-44.</p> <p>“Thus, an ‘information object’ according to the invention may include an applet which, for example, may generate an animated figure or icon, some aural output, a scrolling display, or a combination thereof.” Col. 8, ll. 9-12.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to position the temporary graphic element over the content viewing area to obstruct only part of the content in the content viewing area. Judson ‘643 discloses that “the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above,” e.g., a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area.</p> <p>In addition, the combination of Judson ‘643 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		(see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Judson ‘643 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
36.5	wherein content comprises data for presentation which is from a source external to the browser.	“The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server. In response to the user clicking on the link, the link is activated by the browser to thereby request downloading of the hypertext document from the remote server to the graphical user interface of the client.” Col. 2, ll. 32-39.
37	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The ‘World Wide Web’ (WWW) is that collection of servers of the Internet that utilize the Hypertext Transfer Protocol (HTTP). HTTP is a known application protocol that provides users access to files (which can be in different formats such as text, graphics, images, sound, video, etc.) using a standard page description language known as Hypertext Markup Language (HTML). HTML provides basic document formatting and allows the developer to specify “links” to other servers and files. Use of an HTML-compliant client browser involves specification of a link via the URL. Upon such specification, the client makes a tcp/ip request to the server identified in the link and receives a ‘web page’ (namely, a document formatted according to HTML) in return.” Col. 3, l. 58-Col. 4, l. 4.</p> <p>“RAM 34 also supports a number of Internet access tools including, for example, the HTTP-compliant web browser 62. Known browser software includes Netscape, Netscape Navigator 2.0, Mosaic, and the like. The present invention is designed to operate within any of these known or developing web browsers, which are preferably</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		modified as described herein to achieve the dynamic display of information objects during web site linking activities.” Col. 4, ll. 39-47. “As noted above, a web browser 62 running on the client uses a TCP/IP connection to pass a request to a web server running a HTTP “service” (under the WINDOWS operating system) or “daemon” (under the UNIX operating system). The HTTP service then responds to the request, typically by sending a “web page” formatted in the Hypertext Markup Language, or HTML, to the browser. The browser then displays the web page using local resources (e.g., fonts and colors).” Col. 5, ll. 41-49.
38	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>“The ‘World Wide Web’ (WWW) is that collection of servers of the Internet that utilize the Hypertext Transfer Protocol (HTTP). HTTP is a known application protocol that provides users access to files (which can be in different formats such as text, graphics, images, sound, video, etc.) using a standard page description language known as Hypertext Markup Language (HTML). HTML provides basic document formatting and allows the developer to specify “links” to other servers and files. Use of an HTML-compliant client browser involves specification of a link via the URL. Upon such specification, the client makes a tcp/ip request to the server identified in the link and receives a ‘web page’ (namely, a document formatted according to HTML) in return.” Col. 3, l. 58-Col. 4, l. 4.</p> <p>“RAM 34 also supports a number of Internet access tools including, for example, the HTTP-compliant web browser 62. Known browser software includes Netscape, Netscape Navigator 2.0, Mosaic, and the like. The present invention is designed to operate within any of these known or developing web browsers, which are preferably modified as described herein to achieve the dynamic display of information objects during web site linking activities.” Col. 4, ll. 39-47.</p> <p>“As noted above, a web browser 62 running on the client uses a TCP/IP connection to pass a request to a web server running a HTTP “service” (under the WINDOWS operating system) or “daemon” (under the UNIX operating system). The HTTP service then responds to the request, typically by sending a “web page” formatted in the Hypertext Markup Language, or HTML, to the browser. The browser then</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
39	A computer-readable medium as recited in claim 36 further having additional computer-executable instructions that perform a method comprising, upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	displays the web page using local resources (e.g., fonts and colors).” Col. 5, ll. 41-49. “While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract. “While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to upon completion of the loading, remove the “load status” graphic element. Moreover, Vaughton ‘744 indicates, e.g., that “Task B destroys the window (4) for Task B, when the processing for Task B is complete.” Vaughton ‘744 at Col. 4, ll. 1-2. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40	An information processing device comprising:	“A method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser.” Abstract.
40.1	a processor;	“The microprocessor 32 is connected to the system bus 31 and is supported by read only memory (ROM) 33 and random access memory (RAM) 34 also connected to system bus 31.” Col. 4, ll. 10-13.
40.2	a display;	“The keyboard controller 38 provides the hardware interface for the keyboard 22, the controller 39 provides the hardware interface for the mouse (or other point and click device) 23, the video controller 40 is the hardware interface for the display 24, and the audio controller 41 is the hardware interface for the multimedia speakers 25a and 25b.” Col. 4, ll. 27-33.

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
40.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser is configured to operate in a content-loading mode and a content-loaded mode;	<p>“A method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser. The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server.” Abstract.</p> <p>“According to the preferred embodiment, there is described a method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser.” Col. 2, ll. 29-32.</p> <p>“Each of the clients may run a ‘browser,’ which is a known software tool used to access the servers via the access providers.” Col. 3, ll. 51-53.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p>
40.4	in the content-loaded mode, the hypermedia browser displays loaded content in the content viewing area and no “load status” graphic element is displayed, wherein absence of such “load status” graphic element indicates that the browser is in the content-loaded mode;	<p>“A method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser. The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server.” Abstract.</p> <p>“According to the preferred embodiment, there is described a method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser.” Col. 2, ll. 29-32.</p> <p>“Each of the clients may run a ‘browser,’ which is a known software tool used to access the servers via the access providers.” Col. 3, ll. 51-53.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser, “the screen being without a ‘load status’ graphic element.” Judson ‘643 discloses that “one of ordinary skill in the art would also recognize that [the invention] may be carried out in hardware, in firmware, or in more specialized apparatus constructed to perform the required method steps,” e.g., to save space on a display with a limited display area. Based on this, one of skill in the art would have been motivated to implement a browser having a screen without a “load status” graphic element. Additionally, having a screen without a “load status” graphic element is an obvious design choice where display area is limited and would have been obvious to try. See claim 1 above.</p>
40.5	<p>in the content-loading mode, the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a “load status” graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such “load status” graphic element indicates that the browser is in the content-loading mode; and</p>	<p>“A method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser. The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server.” Abstract.</p> <p>“According to the preferred embodiment, there is described a method of browsing the Worldwide Web of the Internet using an HTML-compliant client supporting a graphical user interface and a browser.” Col. 2, ll. 29-32.</p> <p>“Each of the clients may run a ‘browser,’ which is a known software tool used to access the servers via the access providers.” Col. 3, ll. 51-53.</p> <p>“While the client waits for a reply and/or as the hypertext document is being downloaded, the browser displays one or more different types of informational messages to the user.” Abstract.</p> <p>“While the client waits for a reply and/or as the hypertext document is being</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643’”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744’”) and/or various systems implementing a web browser on a mobile device
		<p>downloaded, the browser displays one or more different types of informational messages to the user.” Col. 2, ll. 39-42.</p> <p>“FIG. 8 is a representative screen display illustrating how the information object appears as a ‘mini’ web page upon activation of the hypertext link in the web page of FIG. 5.” Col. 3, ll. 38-40.</p> <p>“As used herein, the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above.” Col. 7, ll. 39-44.</p> <p>“Thus, an ‘information object’ according to the invention may include an applet which, for example, may generate an animated figure or icon, some aural output, a scrolling display, or a combination thereof.” Col. 8, ll. 9-12.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to position the temporary graphic element over the content viewing area to obstruct only part of the content in the content viewing area. Judson ‘643 discloses that “the ‘information object’ or ‘information’ output to the viewer during the link process should be broadly construed to cover any and all forms of messages, notices, text, graphics, sound, video, tables, diagrams, applets and other content, and combinations of any of the above,” e.g., a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area.</p> <p>In addition, the combination of Judson ‘643 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Judson ‘643 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40.6	wherein content comprises data for presentation which is from a source external to the browser.	“The method begins as a web page is being displayed on the graphical user interface, the web page having at least one link to a hypertext document preferably located at a remote server. In response to the user clicking on the link, the link is activated by the browser to thereby request downloading of the hypertext document from the remote server to the graphical user interface of the client.” Col. 2, ll. 32-39.
41	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The ‘World Wide Web’ (WWW) is that collection of servers of the Internet that utilize the Hypertext Transfer Protocol (HTTP). HTTP is a known application protocol that provides users access to files (which can be in different formats such as text, graphics, images, sound, video, etc.) using a standard page description language known as Hypertext Markup Language (HTML). HTML provides basic document formatting and allows the developer to specify “links” to other servers and files. Use of an HTML-compliant client browser involves specification of a link via the URL. Upon such specification, the client makes a tcp/ip request to the server identified in the link and receives a ‘web page’ (namely, a document formatted according to HTML) in return.” Col. 3, l. 58-Col. 4, l. 4.</p> <p>“RAM 34 also supports a number of Internet access tools including, for example, the HTTP-compliant web browser 62. Known browser software includes Netscape, Netscape Navigator 2.0, Mosaic, and the like. The present invention is designed to operate within any of these known or developing web browsers, which are preferably modified as described herein to achieve the dynamic display of information objects during web site linking activities.” Col. 4, ll. 39-47.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,572,643 (“Judson ‘643”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>“As noted above, a web browser 62 running on the client uses a TCP/IP connection to pass a request to a web server running a HTTP “service” (under the WINDOWS operating system) or “daemon” (under the UNIX operating system). The HTTP service then responds to the request, typically by sending a “web page” formatted in the Hypertext Markup Language, or HTML, to the browser. The browser then displays the web page using local resources (e.g., fonts and colors).” Col. 5, ll. 41-49.</p>
42	<p>A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.</p>	<p>“The ‘World Wide Web’ (WWW) is that collection of servers of the Internet that utilize the Hypertext Transfer Protocol (HTTP). HTTP is a known application protocol that provides users access to files (which can be in different formats such as text, graphics, images, sound, video, etc.) using a standard page description language known as Hypertext Markup Language (HTML). HTML provides basic document formatting and allows the developer to specify “links” to other servers and files. Use of an HTML-compliant client browser involves specification of a link via the URL. Upon such specification, the client makes a tcp/ip request to the server identified in the link and receives a ‘web page’ (namely, a document formatted according to HTML) in return.” Col. 3, l. 58-Col. 4, l. 4.</p> <p>“RAM 34 also supports a number of Internet access tools including, for example, the HTTP-compliant web browser 62. Known browser software includes Netscape, Netscape Navigator 2.0, Mosaic, and the like. The present invention is designed to operate within any of these known or developing web browsers, which are preferably modified as described herein to achieve the dynamic display of information objects during web site linking activities.” Col. 4, ll. 39-47.</p> <p>“As noted above, a web browser 62 running on the client uses a TCP/IP connection to pass a request to a web server running a HTTP “service” (under the WINDOWS operating system) or “daemon” (under the UNIX operating system). The HTTP service then responds to the request, typically by sending a “web page” formatted in the Hypertext Markup Language, or HTML, to the browser. The browser then displays the web page using local resources (e.g., fonts and colors).” Col. 5, ll. 41-49.</p>

EXHIBIT C

Exemplar Chart of U.S. Patent 6,339,780**U.S. Patent 6,023,698 (“Lavey ‘698”)¹
Claims 1-6, 9-14, 17, 18, 20, 21 and 32-42**

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
1	A hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area,	<p>“FIG. 2B shows a schematic block diagram of a client computer 30 running a client/server application according to the present invention connected to a server site 40 on the Internet 37. Client computer 30 includes a processor 31 connected to each of a memory 32, a storage device and/or CD-ROM 33, an input device 34 and a display 35 in a well-known manner. Memory 32 includes instructions and information for the client application.” Col. 5, ll. 11-18.</p> <p>“For the second type of online resource, the application launches a web browser as a foreground process for accessing a specific online server site. When the web browser is launched, a uniform resource locator (URL) information and any other required connection parameters are passed for connecting with the online resource.” Col. 5, ll. 62-65.</p> <p>To the extent a hypermedia browser is found not to be expressly or inherently disclosed, it would have been obvious to one of skill in the art to adapt the “download status bar” described in Lavey ‘698 to visual browsing.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser on an information processing device having</p>

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “System and Method for Transparently Registering and Updating Information Over the Internet” issued to Lavey, Jr.; filed December 5, 1996; issued February 8, 2000.

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>a limited display area.</p> <p>In addition, the combination of Lavey ‘698 and one or more references describing handheld devices with browsers renders this claim obvious. Gessler², Cooper³, Bartlett⁴, Watson⁵, Kamba⁶, and Lauff⁷ each teach a hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area (e.g., a PDA, handheld, mobile device, etc.), and it would be obvious to one of skill in the art to modify Lavey ‘698 with these teachings.⁸ One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Additionally, Bartlett cites Gessler and Cooper cites Gessler and Bartlett. Moreover, all of these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

² S. Gessler and A. Kotulla. PDAs as Mobile WWW Browsers. Computer Networks and ISDN Systems 28 (1995) 53-59 (“Gessler”) (see pgs. 53 and 56-58).

³ I. Cooper and R. Shufflebotham. PDWeb Browsers: Implementation Issues. (University of Kent at Canterbury, November 9, 1995) (“Cooper”) (see pgs. 3 and 5).

⁴ J. Bartlett. Experience with a Wireless World Wide Web Client. IEEE COMPCON 95 (San Francisco, March 1995) (“Bartlett”) (see pgs. 1, 4 and 5).

⁵ T. Watson. “Application Design for Wireless Computing” (1994 Mobile Computing Systems and Applications Workshop Position Paper (August 1994)).

⁶ T. Kamba, S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. (1996). “Using small screen space more efficiently.”

⁷ M. Lauff and W. Gellerson. Multimedia Client Implementation on Personal Digital Assistants. TecO.

⁸ Motorola continues to investigate other browsers that were in existence prior to the priority date of the ‘780 patent (e.g., NetHopper, Newt’s Cape, PocketWeb and Vagon).

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 ("Lavey '698") alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744") and/or various systems implementing a web browser on a mobile device
1.1	wherein the hypermedia browser has a content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing area during times when the browser is loading content,	<p>"Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer." Col. 2, ll. 62-64.</p> <p>"FIG. 2C shows an exemplary graphical display of status of an object request." Col. 3, ll. 22-23.</p> <p>"All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user." Col. 6, ll. 14-23.</p> <div data-bbox="776 655 1250 1264"> <p>FIG. 2C is a diagram of a graphical display 45, which is a PC desktop. It contains a window 47 labeled "(CLIENT APPLICATION)". Inside this window is a "DOWNLOAD STATUS BAR" 46. The status bar features a horizontal progress bar with a shaded portion indicating 35% completion. Numerical markers "0", "50", and "100" are positioned along the bottom of the progress bar. To the right of the progress bar, the text "PERCENT COMPLETE" is displayed above the value "35%".</p> </div>

FIG. 2C

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
1.2	wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area,	See FIG. 2C where exemplary status bar 46 only covers a portion of the “(client application)” on the “graphical display (i.e., PC desktop).”
1.3	wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser.	“Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 20-23.
2	A hypermedia browser as recited in claim 1, wherein the browser is configured to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C.</p>
3	A hypermedia browser as recited in claim 1, wherein the temporary graphic element indicates to a user that the browser is loading content.	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C.</p>

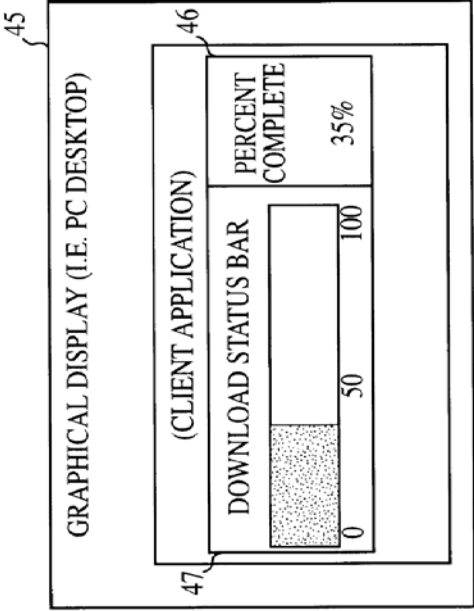
Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
4	A hypermedia browser as recited in claim 1, wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete. Lavey ‘698 discloses that “a status bar and/or an animated icon appear[s] in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer.” Based on this, one of skill in the art would have been motivated to implement a browser wherein the status bar and/or animated icon disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete.</p> <p>In addition, the combination of Lavey ‘698 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Lavey ‘698 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area),</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
5	A hypermedia browser as recited in claim 1, wherein the temporary graphic element is animated.	“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C.
6	A hypermedia browser as recited in claim 1, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area. Lavey ‘698 discloses that “a status bar and/or an animated icon appear[s] in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer.” Based on this, one of skill in the art would have been motivated to implement a browser wherein the status bar and/or animated icon is displayed in a corner of the content viewing area. Additionally, displaying the status bar and/or animated icon in a corner of the content viewing area is an obvious design choice to minimize the amount of content obstructed and would have been obvious to try. Moreover, Vaughton ‘744 indicates, e.g., that “the window(s) of the second task may abut the edge of the window(s) of the first task--the windows may be “clipped together” at their edge.” Vaughton ‘744 at Col. 2, l. 63 - Col. 3, l. 3. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period.

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
9	A hypermedia browser as recited in claim 1, wherein the temporary graphic element conveys status information of the browser.	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C.</p>
10	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The transparent interface protocol of the present invention permits convenient interaction with an online server site, such as for receiving requested information, receiving any new information necessary for operation of the application, for logging any information transfer, and for closing the connection when desired.” Col. 3, ll. 59-65.</p> <p>“For the present invention, this layer of the model preferably conforms to the Hypertext Transfer Protocol (HTTP). The TRUE/IP protocol of the present invention also supports any other IP-based protocol, such as File Transfer Protocol (FTP), Mail, Telnet, Gopher, Network News Transport Protocol (NNTP), Chat, Forums, etc.” Col. 4, ll. 39-44.</p> <p>“In FIG. 2A, a client/server application 20 accesses an online server site using TRUE/IP with or without a browser 20a, containing information, such as any of types 21, 24 or 25, over the Internet 22 through an Internet service provider (ISP) 23.” Col. 4, ll. 61-66.</p> <p>“Information types 21, 24 and 25 available on the online server each are accessible</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		over the Internet and can include a host computer running a Hypertext Transfer Protocol (HTTP) server, such as an IBM™ Internet Connection Server.” Col. 5, ll. 2-6.
11	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>“The transparent interface protocol of the present invention permits convenient interaction with an online server site, such as for receiving requested information, receiving any new information necessary for operation of the application, for logging any information transfer, and for closing the connection when desired.” Col. 3, ll. 59-65.</p> <p>“For the present invention, this layer of the model preferably conforms to the Hypertext Transfer Protocol (HTTP). The TRUE/IP protocol of the present invention also supports any other IP-based protocol, such as File Transfer Protocol (FTP), Mail, Telnet, Gopher, Network News Transport Protocol (NNTP), Chat, Forums, etc.” Col. 4, ll. 39-44.</p> <p>“In FIG. 2A, a client/server application 20 accesses an online server site using TRUE/IP with or without a browser 20a, containing information, such as any of types 21, 24 or 25, over the Internet 22 through an Internet service provider (ISP) 23.” Col. 4, ll. 61-66.</p> <p>“Information types 21, 24 and 25 available on the online server each are accessible over the Internet and can include a host computer running a Hypertext Transfer Protocol (HTTP) server, such as an IBM™ Internet Connection Server.” Col. 5, ll. 2-6.</p> <p>“A system and method for retrieving information from an online database.” Abstract.</p>
12	An information processing device comprising:	
12.1	a processor;	“A client computer that communicates with the server computer includes a client memory and a client processor. The client memory, which can include a storage device and/or CD-ROM, stores client application instructions that include a set of dynamic link libraries of code and information for each of a plurality of Internet service providers. The client processor is connected to the client memory and is responsive to the client application instructions by establishing a connection with the

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>server computer over the Internet through a selected Internet service provider and sending the token to the server computer.” Col. 2, ll. 12-22.</p> <p>“FIG. 2B shows a schematic block diagram of a client computer 30 running a client/server application according to the present invention connected to a server site 40 on the Internet 37. Client computer 30 includes a processor 31 connected to each of a memory 32, a storage device and/or CD-ROM 33, an input device 34 and a display 35 in a well-known manner. Memory 32 includes instructions and information for the client application.” Col. 5, ll. 11-18.</p>
12.2	a display;	<p>“FIG. 2B shows a schematic block diagram of a client computer 30 running a client/server application according to the present invention connected to a server site 40 on the Internet 37. Client computer 30 includes a processor 31 connected to each of a memory 32, a storage device and/or CD-ROM 33, an input device 34 and a display 35 in a well-known manner. Memory 32 includes instructions and information for the client application.” Col. 5, ll. 11-18.</p> <p>“Display 35 provides a visual interface to client computer 30 in a well-known manner, such as by providing a graphical user interface (GUI) (FIG. 2C) in accordance with instructions and information stored in memory 32 and storage device and/or CD-ROM 33 for the client application.” Col. 5, ll. 29-33.</p>
12.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser has a content viewing area for viewing content and wherein the hypermedia browser displays a temporary graphic element over the content viewing area during times when the browser is loading visible content;	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“FIG. 2C shows an exemplary graphical display of status of an object request.” Col. 3, ll. 22-23.</p> <p>“For the second type of online resource, the application launches a web browser as a foreground process for accessing a specific online server site. When the web browser is launched, a uniform resource locator (URL) information and any other required connection parameters are passed for connecting with the online resource.” Col. 5, ll. 62-65.</p>

Claim	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device	<p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23.</p>  <p style="text-align: center;">FIG. 2C</p> <p>To the extent a hypermedia browser is found not to be expressly or inherently disclosed, it would have been obvious to one of skill in the art to adapt the “download status bar” described in Lavey ‘698 to visual browsing.</p>
12.4	wherein the temporary graphic element is positioned only over a portion of the content viewing area and obstructs only	See FIG. 2C where exemplary status bar 46 only covers a portion of the “(client application)” on the “graphical display (i.e., PC desktop).”

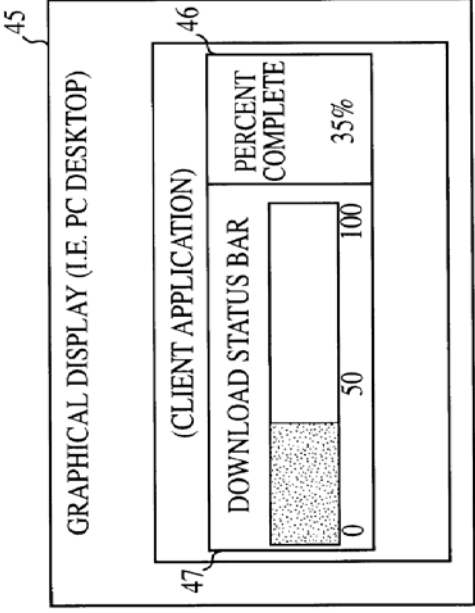
Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
	part of the visible content in the content viewing area; and	
12.5	wherein the temporary graphic element indicates to a user that the browser is loading content and content comprises data for presentation which is from a source external to the browser.	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C.</p>
13	An information processing device as recited in claim 12, wherein the temporary graphic element is animated.	<p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C.</p>
14	An information processing device as recited in claim 12, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area. Lavey ‘698 discloses that “a status bar and/or an animated icon appear[s] in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer.” Based on this, one of skill in the art would have been motivated to implement a browser wherein the status bar and/or animated icon is displayed in a corner of the content viewing area. Additionally, displaying the status bar and/or animated icon in a corner of the content viewing area is an obvious design choice to minimize the amount of content</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		obstructed and would have been obvious to try. Moreover, Vaughton ‘744 indicates, e.g., that “the window(s) of the second task may about the edge of the window(s) of the first task--the windows may be “clipped together” at their edge.” Vaughton ‘744 at Col. 2, l. 63 - Col. 3, l. 3. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Moreover, these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
17	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The transparent interface protocol of the present invention permits convenient interaction with an online server site, such as for receiving requested information, receiving any new information necessary for operation of the application, for logging any information transfer, and for closing the connection when desired.” Col. 3, ll. 59-65.</p> <p>“For the present invention, this layer of the model preferably conforms to the Hypertext Transfer Protocol (HTTP). The TRUE/IP protocol of the present invention also supports any other IP-based protocol, such as File Transfer Protocol (FTP), Mail, Telnet, Gopher, Network News Transport Protocol (NNTP), Chat, Forums, etc.” Col. 4, ll. 39-44.</p> <p>“In FIG. 2A, a client/server application 20 accesses an online server site using TRUE/IP with or without a browser 20a, containing information, such as any of types 21, 24 or 25, over the Internet 22 through an Internet service provider (ISP) 23.” Col. 4, ll. 61-66.</p> <p>“Information types 21, 24 and 25 available on the online server each are accessible over the Internet and can include a host computer running a Hypertext Transfer Protocol (HTTP) server, such as an IBMTM Internet Connection Server.” Col. 5, ll. 2-6.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
18	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>“The transparent interface protocol of the present invention permits convenient interaction with an online server site, such as for receiving requested information, receiving any new information necessary for operation of the application, for logging any information transfer, and for closing the connection when desired.” Col. 3, ll. 59-65.</p> <p>“For the present invention, this layer of the model preferably conforms to the Hypertext Transfer Protocol (HTTP). The TRUE/IP protocol of the present invention also supports any other IP-based protocol, such as File Transfer Protocol (FTP), Mail, Telnet, Gopher, Network News Transport Protocol (NNTP), Chat, Forums, etc.” Col. 4, ll. 39-44.</p> <p>“In FIG. 2A, a client/server application 20 accesses an online server site using TRUE/IP with or without a browser 20a, containing information, such as any of types 21, 24 or 25, over the Internet 22 through an Internet service provider (ISP) 23.” Col. 4, ll. 61-66.</p> <p>“Information types 21, 24 and 25 available on the online server each are accessible over the Internet and can include a host computer running a Hypertext Transfer Protocol (HTTP) server, such as an IBMTM Internet Connection Server.” Col. 5, ll. 2-6.</p>
20	An information processing device as recited in claim 12, wherein the temporary graphic element is not content.	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C.</p>
21	An information processing device as	“Preferably, the client computer graphically displays the status of the object requested

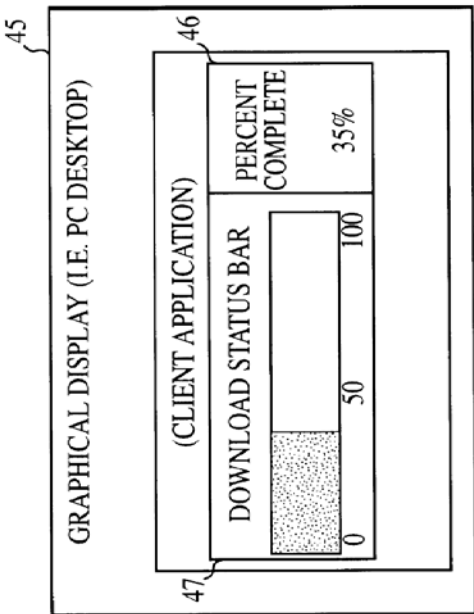
Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
	<p>recited in claim 12, wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.</p>	<p>while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete. Lavey ‘698 discloses that “a status bar and/or an animated icon appear[s] in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer.” Based on this, one of skill in the art would have been motivated to implement a browser wherein the status bar and/or animated icon disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete.</p> <p>In addition, the combination of Lavey ‘698 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Lavey ‘698 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
32	A method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content, the method comprising:	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“FIG. 2C shows an exemplary graphical display of status of an object request.” Col. 3, ll. 22-23.</p> <p>“For the second type of online resource, the application launches a web browser as a foreground process for accessing a specific online server site. When the web browser is launched, a uniform resource locator (URL) information and any other required connection parameters are passed for connecting with the online resource.” Col. 5, ll. 62-65.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 ("Lavey '698") alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744") and/or various systems implementing a web browser on a mobile device
		 <p style="text-align: center;">FIG. 2C</p> <p>To the extent a hypermedia browser is found not to be expressly or inherently disclosed, it would have been obvious to one of skill in the art to adapt the "download status bar" described in Lavey '698 to visual browsing.</p>
32.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen being without a "load status" graphic element, wherein a "load status" graphic element indicates a current content load status of the hypermedia browser;	<p>"Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer." Col. 2, ll. 62-64.</p> <p>"FIG. 2C shows an exemplary graphical display of status of an object request." Col. 3, ll. 22-23.</p> <p>"For the second type of online resource, the application launches a web browser as a foreground process for accessing a specific online server site. When the web browser is launched, a uniform resource locator (URL) information and any other required connection parameters are passed for connecting with the online resource." Col. 5,</p>

Claim	U.S. Patent 6,339,780	<p>U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device</p>
		<p>Il. 62-65.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, Il. 14-23.</p> <div data-bbox="617 657 1088 1264"> <p>FIG. 2C is a schematic diagram of a graphical display 45, which is identified as a PC desktop. Within this display is a window 47 labeled '(CLIENT APPLICATION)'. Inside window 47 is a 'DOWNLOAD STATUS BAR' 46. This bar features a progress indicator consisting of a horizontal bar with a shaded portion representing 35% completion. Numerical markers '0', '50', and '100' are positioned along the bottom of the bar. To the right of the shaded area, the text 'PERCENT COMPLETE' is displayed above the value '35%'.</p> </div> <p style="text-align: center;">FIG. 2C</p>
32.2	receiving an instruction to load new content into the content viewing area;	<p>“Input device 34, such as a keyboard and/or a mouse, accepts user inputs that are processed by processor 31 in accordance with the instructions and information for the client application stored in memory 32 and storage device and/or CD-ROM 33.” Col. 5, Il. 25-29.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
32.3	loading such new content into the content viewing area; and	<p>“Processor 41 uses the instructions and information stored in memory 42 to operate on received tokens in accordance with the appropriate token handler. For example, when an object request token is received and validated by the appropriate token handlers, processor 41 accesses database 43 in a well-known manner for retrieving the requested objected.” Col. 5, ll. 45-51.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23.</p>
32.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“FIG. 2C shows an exemplary graphical display of status of an object request.” Col. 3, ll. 22-23.</p> <p>“For the second type of online resource, the application launches a web browser as a foreground process for accessing a specific online server site. When the web browser is launched, a uniform resource locator (URL) information and any other required connection parameters are passed for connecting with the online resource.” Col. 5, ll. 62-65.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23.</p>

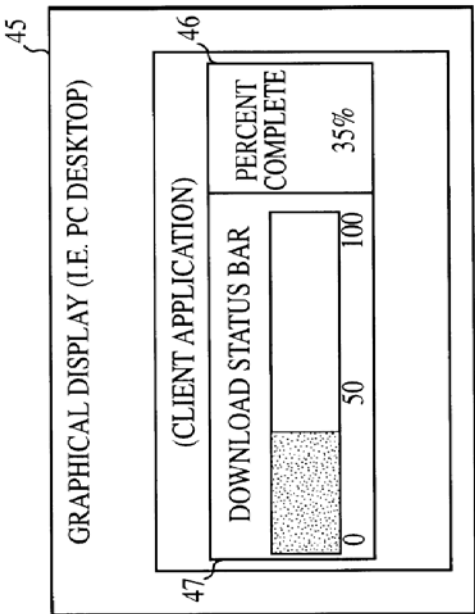
Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		 <p style="text-align: center;">FIG. 2C</p>
32.5	wherein content comprises data for presentation which is from a source external to the browser.	“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.
33	A method as recited in claim 32 further comprising, upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete. Lavey ‘698 discloses that “a status bar and/or an animated icon appear[s] in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer.” Based on this, one of skill in the art would have been motivated to implement a browser wherein the status bar and/or animated icon disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete. In addition, the combination of Lavey ‘698 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Lavey ‘698 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
34	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a	“The transparent interface protocol of the present invention permits convenient interaction with an online server site, such as for receiving requested information, receiving any new information necessary for operation of the application, for logging

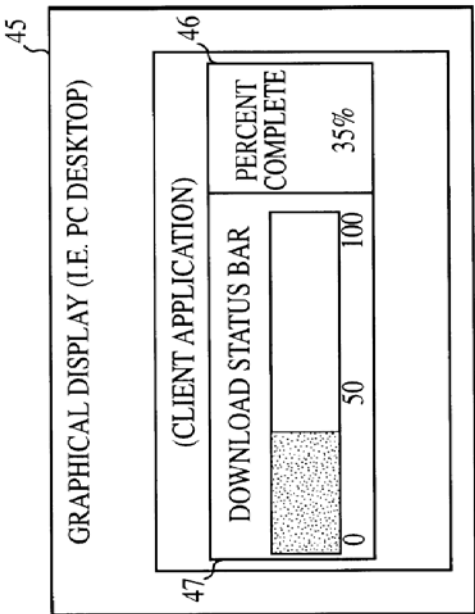
Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
	group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>any information transfer, and for closing the connection when desired.” Col. 3, ll. 59-65.</p> <p>“For the present invention, this layer of the model preferably conforms to the Hypertext Transfer Protocol (HTTP). The TRUE/IP protocol of the present invention also supports any other IP-based protocol, such as File Transfer Protocol (FTP), Mail, Telnet, Gopher, Network News Transport Protocol (NNTP), Chat, Forums, etc.” Col. 4, ll. 39-44.</p> <p>“In FIG. 2A, a client/server application 20 accesses an online server site using TRUE/IP with or without a browser 20a, containing information, such as any of types 21, 24 or 25, over the Internet 22 through an Internet service provider (ISP) 23.” Col. 4, ll. 61-66.</p> <p>“Information types 21, 24 and 25 available on the online server each are accessible over the Internet and can include a host computer running a Hypertext Transfer Protocol (HTTP) server, such as an IBMTM Internet Connection Server.” Col. 5, ll. 2-6.</p>
35	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>“The transparent interface protocol of the present invention permits convenient interaction with an online server site, such as for receiving requested information, receiving any new information necessary for operation of the application, for logging any information transfer, and for closing the connection when desired.” Col. 3, ll. 59-65.</p> <p>“For the present invention, this layer of the model preferably conforms to the Hypertext Transfer Protocol (HTTP). The TRUE/IP protocol of the present invention also supports any other IP-based protocol, such as File Transfer Protocol (FTP), Mail, Telnet, Gopher, Network News Transport Protocol (NNTP), Chat, Forums, etc.” Col. 4, ll. 39-44.</p> <p>“In FIG. 2A, a client/server application 20 accesses an online server site using TRUE/IP with or without a browser 20a, containing information, such as any of types 21, 24 or 25, over the Internet 22 through an Internet service provider (ISP) 23.”</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		Col. 4, ll. 61-66. “Information types 21, 24 and 25 available on the online server each are accessible over the Internet and can include a host computer running a Hypertext Transfer Protocol (HTTP) server, such as an IBM TM Internet Connection Server.” Col. 5, ll. 2-6.
36	A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content, the method comprising:	<p>“A client computer includes a client memory that stores client application instructions and a client processor connected to the client processor. The client application instructions include a set of dynamic link libraries of code and information for each of a plurality of Internet service providers. The client processor is responsive to the client application instructions by establishing a connection with the server computer over the Internet through a selected Internet service provider and by sending tokens to the server computer.” Abstract.</p> <p>“A client computer that communicates with the server computer includes a client memory and a client processor. The client memory, which can include a storage device and/or CD-ROM, stores client application instructions that include a set of dynamic link libraries of code and information for each of a plurality of Internet service providers. The client processor is connected to the client memory and is responsive to the client application instructions by establishing a connection with the server computer over the Internet through a selected Internet service provider and sending the token to the server computer.” Col. 2, ll. 12-22.</p> <p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“FIG. 2B shows a schematic block diagram of a client computer 30 running a client/server application according to the present invention connected to a server site 40 on the Internet 37. Client computer 30 includes a processor 31 connected to each of a memory 32, a storage device and/or CD-ROM 33, an input device 34 and a display 35 in a well-known manner. Memory 32 includes instructions and information for the client application.” Col. 5, ll. 11-18.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		<p>“For the second type of online resource, the application launches a web browser as a foreground process for accessing a specific online server site. When the web browser is launched, a uniform resource locator (URL) information and any other required connection parameters are passed for connecting with the online resource.” Col. 5, ll. 62-65.</p> <p>To the extent a hypermedia browser is found not to be expressly or inherently disclosed, it would have been obvious to one of skill in the art to adapt the “download status bar” described in Lavey ‘698 to visual browsing.</p>
36.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen is without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“FIG. 2C shows an exemplary graphical display of status of an object request.” Col. 3, ll. 22-23.</p> <p>“For the second type of online resource, the application launches a web browser as a foreground process for accessing a specific online server site. When the web browser is launched, a uniform resource locator (URL) information and any other required connection parameters are passed for connecting with the online resource.” Col. 5, ll. 62-65.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23.</p>

Claim	U.S. Patent 6,023,698 ("Lavey '698") alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744") and/or various systems implementing a web browser on a mobile device	 <p style="text-align: center;">FIG. 2C</p>
36.2	receiving an instruction to load new content into the content viewing area;	<p>"Input device 34, such as a keyboard and/or a mouse, accepts user inputs that are processed by processor 31 in accordance with the instructions and information for the client application stored in memory 32 and storage device and/or CD-ROM 33."</p> <p>Col. 5, ll. 25-29.</p> <p>"Processor 41 uses the instructions and information stored in memory 42 to operate on received tokens in accordance with the appropriate token handler. For example, when an object request token is received and validated by the appropriate token handlers, processor 41 accesses database 43 in a well-known manner for retrieving the requested objected." Col. 5, ll. 45-51.</p>
36.3	loading such new content into the content viewing area; and	<p>"All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23.
36.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	<p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“FIG. 2C shows an exemplary graphical display of status of an object request.” Col. 3, ll. 22-23.</p> <p>“For the second type of online resource, the application launches a web browser as a foreground process for accessing a specific online server site. When the web browser is launched, a uniform resource locator (URL) information and any other required connection parameters are passed for connecting with the online resource.” Col. 5, ll. 62-65.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23.</p>

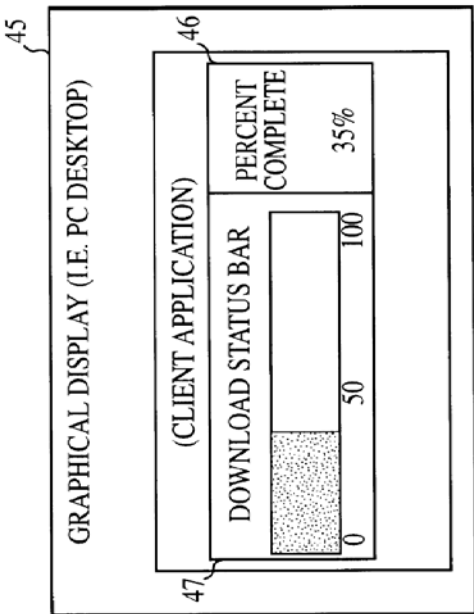
Claim	U.S. Patent 6,023,698 ("Lavey '698") alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744") and/or various systems implementing a web browser on a mobile device	 <p style="text-align: center;">FIG. 2C</p>
36.5	wherein content comprises data for presentation which is from a source external to the browser.	<p>"Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer." Col. 2, ll. 62-64.</p>
37	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>"The transparent interface protocol of the present invention permits convenient interaction with an online server site, such as for receiving requested information, receiving any new information necessary for operation of the application, for logging any information transfer, and for closing the connection when desired." Col. 3, ll. 59-65.</p> <p>"For the present invention, this layer of the model preferably conforms to the Hypertext Transfer Protocol (HTTP). The TRUE/IP protocol of the present invention also supports any other IP-based protocol, such as File Transfer Protocol (FTP), Mail, Telnet, Gopher, Network News Transfer Protocol (NNTP), Chat, Forums, etc."</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		Col. 4, ll. 39-44. <p>“In FIG. 2A, a client/server application 20 accesses an online server site using TRUE/IP with or without a browser 20a, containing information, such as any of types 21, 24 or 25, over the Internet 22 through an Internet service provider (ISP) 23.” Col. 4, ll. 61-66.</p> <p>“Information types 21, 24 and 25 available on the online server each are accessible over the Internet and can include a host computer running a Hypertext Transfer Protocol (HTTP) server, such as an IBMTM Internet Connection Server.” Col. 5, ll. 2-6.</p>
38	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>“The transparent interface protocol of the present invention permits convenient interaction with an online server site, such as for receiving requested information, receiving any new information necessary for operation of the application, for logging any information transfer, and for closing the connection when desired.” Col. 3, ll. 59-65.</p> <p>“For the present invention, this layer of the model preferably conforms to the Hypertext Transfer Protocol (HTTP). The TRUE/IP protocol of the present invention also supports any other IP-based protocol, such as File Transfer Protocol (FTP), Mail, Telnet, Gopher, Network News Transport Protocol (NNTP), Chat, Forums, etc.” Col. 4, ll. 39-44.</p> <p>“In FIG. 2A, a client/server application 20 accesses an online server site using TRUE/IP with or without a browser 20a, containing information, such as any of types 21, 24 or 25, over the Internet 22 through an Internet service provider (ISP) 23.” Col. 4, ll. 61-66.</p> <p>“Information types 21, 24 and 25 available on the online server each are accessible over the Internet and can include a host computer running a Hypertext Transfer Protocol (HTTP) server, such as an IBMTM Internet Connection Server.” Col. 5, ll. 2-6.</p>
39	A computer-readable medium as recited	“Preferably, the client computer graphically displays the status of the object requested

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
	<p>in claim 36 further having additional computer-executable instructions that perform a method comprising, upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.</p>	<p>while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete. Lavey ‘698 discloses that “a status bar and/or an animated icon appear[s] in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer.” Based on this, one of skill in the art would have been motivated to implement a browser wherein the status bar and/or animated icon disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete.</p> <p>In addition, the combination of Lavey ‘698 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Lavey ‘698 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator</p>

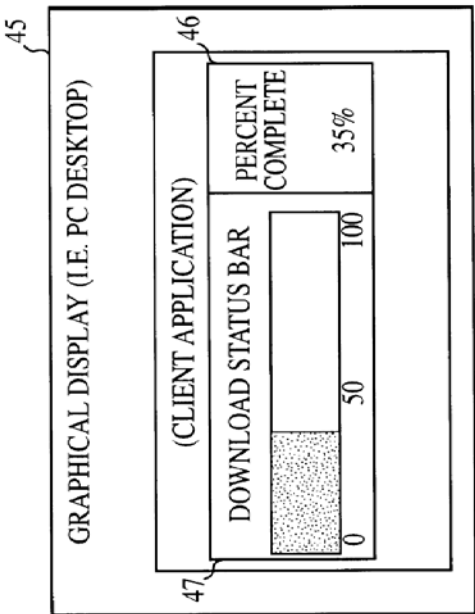
Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
		over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40	An information processing device	“A system and method for retrieving information from an online database.” Abstract.
40.1	comprising: a processor;	<p>“A client computer that communicates with the server computer includes a client memory and a client processor. The client memory, which can include a storage device and/or CD-ROM, stores client application instructions that include a set of dynamic link libraries of code and information for each of a plurality of Internet service providers. The client processor is connected to the client memory and is responsive to the client application instructions by establishing a connection with the server computer over the Internet through a selected Internet service provider and sending the token to the server computer.” Col. 2, ll. 12-22.</p> <p>“FIG. 2B shows a schematic block diagram of a client computer 30 running a client/server application according to the present invention connected to a server site 40 on the Internet 37. Client computer 30 includes a processor 31 connected to each of a memory 32, a storage device and/or CD-ROM 33, an input device 34 and a display 35 in a well-known manner. Memory 32 includes instructions and information for the client application.” Col. 5, ll. 11-18.</p>
40.2	a display;	<p>“FIG. 2B shows a schematic block diagram of a client computer 30 running a client/server application according to the present invention connected to a server site 40 on the Internet 37. Client computer 30 includes a processor 31 connected to each of a memory 32, a storage device and/or CD-ROM 33, an input device 34 and a display 35 in a well-known manner. Memory 32 includes instructions and information for the client application.” Col. 5, ll. 11-18.</p> <p>“Display 35 provides a visual interface to client computer 30 in a well-known manner, such as by providing a graphical user interface (GUI) (FIG. 2C) in accordance with</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device
40.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser is configured to operate in a content-loading mode and a content-loaded mode;	<p>instructions and information stored in memory 32 and storage device and/or CD-ROM 33 for the client application.” Col. 5, ll. 29-33.</p> <p>“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.</p> <p>“FIG. 2C shows an exemplary graphical display of status of an object request.” Col. 3, ll. 22-23.</p> <p>“For the second type of online resource, the application launches a web browser as a foreground process for accessing a specific online server site. When the web browser is launched, a uniform resource locator (URL) information and any other required connection parameters are passed for connecting with the online resource.” Col. 5, ll. 62-65.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 ("Lavey '698") alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744") and/or various systems implementing a web browser on a mobile device
		 <p style="text-align: center;">FIG. 2C</p> <p>To the extent a hypermedia browser is found not to be expressly or inherently disclosed, it would have been obvious to one of skill in the art to adapt the "download status bar" described in Lavey '698 to visual browsing.</p> <p>"Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer." Col. 2, ll. 62-64.</p> <p>"All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data</p>
40.4	in the content-loaded mode, the hypermedia browser displays loaded content in the content viewing area and no "load status" graphic element is displayed, wherein absence of such "load status" graphic element indicates that the browser is in the content-loaded mode;	

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device transfer to a user.” Col. 6, ll. 14-23. <i>See also</i> FIG. 2C. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete. Lavey ‘698 discloses that “a status bar and/or an animated icon appear[s] in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer.” Based on this, one of skill in the art would have been motivated to implement a browser wherein the status bar and/or animated icon disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete. In addition, the combination of Lavey ‘698 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Lavey ‘698 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40.5	in the content-loading mode, the hypermedia browser loads content, displays such content in the content	“Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer.” Col. 2, ll. 62-64.

Claim	<p>U.S. Patent 6,339,780</p> <p>viewing area as it loads, and displays a “load status” graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such “load status” graphic element indicates that the browser is in the content-loading mode; and</p>	<p>U.S. Patent 6,023,698 (“Lavey ‘698”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or various systems implementing a web browser on a mobile device</p> <p>“FIG. 2C shows an exemplary graphical display of status of an object request.” Col. 3, ll. 22-23.</p> <p>“For the second type of online resource, the application launches a web browser as a foreground process for accessing a specific online server site. When the web browser is launched, a uniform resource locator (URL) information and any other required connection parameters are passed for connecting with the online resource.” Col. 5, ll. 62-65.</p> <p>“All other communications, such as data transfers, are reported to a user by a status bar and/or an animated icon appearing in the client applications interface so the user can visually verify that a transfer is occurring, and the approximate percentage of completion of the transfer. FIG. 2C shows an exemplary graphical display 45 that is displayed on display 35 in FIG. 2B. Graphical display 45 includes an exemplary status bar 46 and an exemplary animated icon 47 that communicate the status of a data transfer to a user.” Col. 6, ll. 14-23.</p>
-------	--	---

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 ("Lavey '698") alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744") and/or various systems implementing a web browser on a mobile device
		 <p style="text-align: center;">FIG. 2C</p>
40.6	wherein content comprises data for presentation which is from a source external to the browser.	"Preferably, the client computer graphically displays the status of the object requested while the object is being sent from the host computer to the client computer." Col. 2, ll. 62-64.
41	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>"The transparent interface protocol of the present invention permits convenient interaction with an online server site, such as for receiving requested information, receiving any new information necessary for operation of the application, for logging any information transfer, and for closing the connection when desired." Col. 3, ll. 59-65.</p> <p>"For the present invention, this layer of the model preferably conforms to the Hypertext Transfer Protocol (HTTP). The TRUE/IP protocol of the present invention also supports any other IP-based protocol, such as File Transfer Protocol (FTP), Mail, Telnet, Gopher, Network News Transport Protocol (NNTP), Chat, Forums, etc."</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,023,698 ("Lavey '698") alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744") and/or various systems implementing a web browser on a mobile device
		<p>Col. 4, ll. 39-44.</p> <p>"In FIG. 2A, a client/server application 20 accesses an online server site using TRUE/IP with or without a browser 20a, containing information, such as any of types 21, 24 or 25, over the Internet 22 through an Internet service provider (ISP) 23."</p> <p>Col. 4, ll. 61-66.</p> <p>"Information types 21, 24 and 25 available on the online server each are accessible over the Internet and can include a host computer running a Hypertext Transfer Protocol (HTTP) server, such as an IBMTM Internet Connection Server." Col. 5, ll. 2-6.</p>
42	<p>A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.</p>	<p>"The transparent interface protocol of the present invention permits convenient interaction with an online server site, such as for receiving requested information, receiving any new information necessary for operation of the application, for logging any information transfer, and for closing the connection when desired." Col. 3, ll. 59-65.</p> <p>"For the present invention, this layer of the model preferably conforms to the Hypertext Transfer Protocol (HTTP). The TRUE/IP protocol of the present invention also supports any other IP-based protocol, such as File Transfer Protocol (FTP), Mail, Telnet, Gopher, Network News Transport Protocol (NNTP), Chat, Forums, etc."</p> <p>Col. 4, ll. 39-44.</p> <p>"In FIG. 2A, a client/server application 20 accesses an online server site using TRUE/IP with or without a browser 20a, containing information, such as any of types 21, 24 or 25, over the Internet 22 through an Internet service provider (ISP) 23."</p> <p>Col. 4, ll. 61-66.</p> <p>"Information types 21, 24 and 25 available on the online server each are accessible over the Internet and can include a host computer running a Hypertext Transfer Protocol (HTTP) server, such as an IBMTM Internet Connection Server." Col. 5, ll. 2-6.</p>

EXHIBIT D

Exemplar Chart of U.S. Patent 6,339,780**U.S. Patent 6,584,498 (“Nguyen ‘498”)¹
Claims 1-6, 9-14, 17, 18, 20, 21 and 32-42**

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
1	A hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area,	<p>“The page client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, and program/data memory.” Col. 3, ll. 10-13.</p> <p>“FIG. 1 shows a system for accessing web pages. A system 100 for accessing web pages comprises a page server 110, a communication link 120, and a page client 130.” Col. 2., ll. 36-39. The page client may be a web browser. Col. 1, ll. 52-53. “In a first preferred embodiment, the page client 130 comprises an input element 133 and a display element 134. The input element 133 comprises a keyboard and a pointing device such as a mouse or trackball. The display element 134 comprises a visual display element such as a monitor or a display panel[.]” Col. 3, ll. 22-27.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser on an information processing device having a limited display area.</p> <p>In addition, the combination of Nguyen ‘498 and one or more references describing handheld devices with browsers renders this claim obvious. Gessler², Cooper³,</p>

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Dynamic Loading of Web Pages” issued to Nguyen; filed September 13, 1996; issued June 24, 2003. In the specification, the patentee expressly incorporates his pending application (08/716,641), which later issued as U.S. Patent No. 6,377,978, and deals with the dynamic downloading of hypertext electronic mail messages.

² S. Gessler and A. Kotulla. PDAs as Mobile WWW Browsers. Computer Networks and ISDN Systems 28 (1995) 53-59 (“Gessler”) (see pgs. 53 and 56-58).

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		Bartlett ⁴ , Watson ⁵ , Kamba ⁶ , and Lauff ⁷ each teach a hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area (e.g., a PDA, handheld, mobile device, etc.), and it would be obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. ⁸ One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Additionally, Bartlett cites Gessler and Cooper cites Gessler and Bartlett. Moreover, all of these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.1	wherein the hypermedia browser has a content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing	“A primary page 140 presented on the output element 134 comprises information for presentation to an operator.” Col. 3, ll. 44-45. “The page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator

(...Continued)

³ I. Cooper and R. Shufflebotham. PDWeb Browsers: Implementation Issues. (University of Kent at Canterbury, November 9, 1995) (“Cooper”) (see pgs. 3 and 5).

⁴ J. Bartlett. Experience with a Wireless World Wide Web Client. IEEE COMPCON 95 (San Francisco, March 1995) (“Bartlett”) (see pgs. 1, 4 and 5).

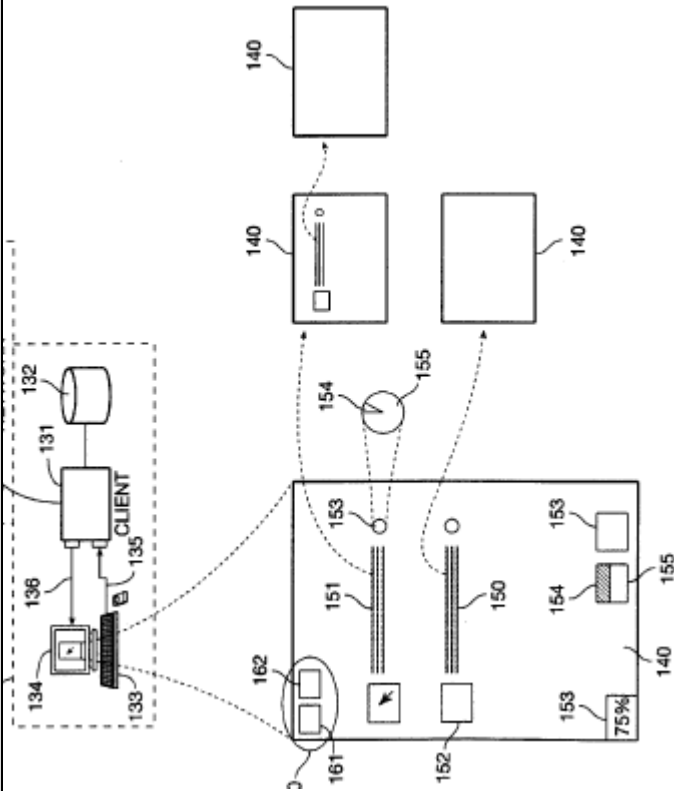
⁵ T. Watson. “Application Design for Wireless Computing” (1994 Mobile Computing Systems and Applications Workshop Position Paper (August 1994)).

⁶ T. Kamba, S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. (1996). “Using small screen space more efficiently.”

⁷ M. Lauff and W. Gellerson. Multimedia Client Implementation on Personal Digital Assistants. TecO.

⁸ Motorola continues to investigate other browsers that were in existence prior to the priority date of the ‘780 patent (e.g., NetHopper, Newt’s Cape, PocketWeb and Vagon).

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
	area during times when the browser is loading content,	153 comprises a first part 154 having a first display color or pattern and a second part 153 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.
1.2	wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area,	“In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 5, ll. 6-8; <i>see also</i> FIG. 1.
1.3	wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser.	<p>“[T]he preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140. In this preferred embodiment, the first part 154 comprises a pie slice of the dot or small circle which by its angle and its color (green) indicates an amount of the “in progress” preloading stage which has been completed, and the second part 155 by its angle and its color (orange) indicates an amount which has not been completed. Thus the image of the dot or small circle will change color from orange to orange/green to green as the preloading operation progresses.” Col. 5, ll. 6-16.</p> <p>Content is provided by a page server external to the browser. <i>See</i> Col. 2, ll. 45-47; <i>see also</i> FIG. 1.</p>

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		 <p>The diagram illustrates a system architecture where a client device interacts with a server. On the left, a dashed box labeled 'CLIENT' contains a computer system with a monitor (134), keyboard (133), mouse (135), and central processing unit (131). A network connection (132) links the client to a server (140) on the right. Below the client, a detailed view of the monitor displays a graphical user interface. This interface features a window (162) showing a document (151) with a cursor (153) positioned over it. At the bottom of the window is a status bar (150) indicating '75%' completion. To the left of the window is a toolbar (154) with icons for various functions, and above it is a menu bar (155). The entire display area is framed by a border (140).</p>
2	A hypermedia browser as recited in claim 1, wherein the browser is configured to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.	<p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 5, ll. 6-8; <i>see also</i> FIG. 1.</p> <p>“The choice of display colors is, of course, completely arbitrary. Other colors, patterns, or other visual or audible indicators may be used instead or in conjunction to present the operator with information about the pre loading stage for any particular link 150.” Col. 4, ll. 53-57.</p>

	U.S. Patent 6,339,780	<p>U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete. Nguyen ‘498 discloses that the preloading indicator indicates an amount of the ‘in progress’ preloading stage which has been completed and an amount which has not been completed. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.</p> <p>In addition, the combination of Nguyen ‘498 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
3	A hypermedia browser as recited in claim 1, wherein the temporary graphic element indicates to a user that the browser is loading content.	<p>“[T]he preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140. In this preferred embodiment, the first part 154 comprises a pie slice of the dot or small circle which by its angle and its color (green) indicates an amount of the ‘in progress’ preloading stage which has been completed, and the</p>

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		second part 155 by its angle and its color (orange) indicates an amount which has not been completed. Thus the image of the dot or small circle will change color from orange to orange/green to green as the preloading operation progresses.” Col. 5, ll. 6-16.
4	A hypermedia browser as recited in claim 1, wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.	<p>“The choice of display colors is, of course, completely arbitrary. Other colors, patterns, or other visual or audible indicators may be used instead or in conjunction to present the operator with information about the pre loading stage for any particular link 150.” Col. 4, ll. 53-57.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete. Nguyen ‘498 discloses that the preloading indicator indicates an amount of the ‘in progress’ preloading stage which has been completed and an amount which has not been completed. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.</p> <p>In addition, the combination of Nguyen ‘498 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator</p>

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
5	A hypermedia browser as recited in claim 1, wherein the temporary graphic element is animated.	“In this preferred embodiment, the first part 154 comprises a pie slice of the dot or small circle which by its angle and its color (green) indicates an amount of the ‘in progress’ preloading stage which has been completed, and the second part 155 by its angle and its color (orange) indicates an amount which has not been completed. Thus the image of the dot or small circle will change color from orange to orange/green to green as the preloading operation progresses.” Col. 5, ll. 8-16.
6	A hypermedia browser as recited in claim 1, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	“In addition to the first part 154 and the second part 155, the preloading indicator 153 comprises a textual or graphical indicator, positioned at a margin of the primary page 140 (such as in a margin for a window used for presentation of the visual elements of the primary page 140), indicating an amount of the “in progress” preloading state which has been completed. For example, when 75% of the secondary page 140 is preloaded, the preloading indicator 153 comprises the text “75% preloaded” or some equivalent statement, or 55 comprises a thermometer graph showing that 75% of the secondary page 140 is preloaded.” Col. 5, ll. 47-57; <i>see also</i> FIG. 1.
9	A hypermedia browser as recited in claim 1, wherein the temporary graphic element conveys status information of the browser.	“In this preferred embodiment, the first part 154 comprises a pie slice of the dot or small circle which by its angle and its color (green) indicates an amount of the ‘in progress’ preloading stage which has been completed, and the second part 155 by its angle and its color (orange) indicates an amount which has not been completed. Thus the image of the dot or small circle will change color from orange to orange/green to green as the preloading operation progresses.” Col. 5, ll. 8-16.

	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device	<p>“In addition to the first part 154 and the second part 155, the preloading indicator 153 comprises a textual or graphical indicator, positioned at a margin of the primary page 140 (such as in a margin for a window used for presentation of 50 the visual elements of the primary page 140), indicating an amount of the “in progress” preloading state which has been completed. For example, when 75% of the secondary page 140 is preloaded, the preloading indicator 153 comprises the text “75% preloaded” or some equivalent statement, or 55 comprises a thermometer graph showing that 75% of the secondary page 140 is preloaded.” Col. 5, ll. 47-57; <i>see also</i> FIG. 1.</p>
10	U.S. Patent 6,339,780	<p>“The page server 110 is disposed for generating and responding to messages in a protocol for presenting web pages to the page client 130. The protocol is preferably the Hypertext Transfer Protocol (“HTTP”), but the concepts of the invention are broad enough to apply to other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols, such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 2, ll. 45-56.</p> <p>“In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 13-15.</p> <p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“Links 150 are embedded in pages 140 using a set of HTML tags, or by other techniques for identifying other pages 140 which may be presented. Links 150 are themselves presented to the operator as text elements 151 or as graphic elements 152. Links 150 may be identified by the page client 130 by special forms of display, for</p>

	U.S. Patent 6,339,780	<p>U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device</p> <p>example by displaying text in a different color or typeface from text elements 151 or as graphic elements 152 which are not associated with links 150.” Col. 4, ll. 17-25.</p> <p>Nguyen ‘498 additionally cites several Java references, e.g., Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996; and Stephen R. Davis. “Teach Yourself Java Programming the Quick and Easy Way with Microsoft Visual: Learn Java Now”. Microsoft Press. 3 0402 00136 8093.</p>
11	<p>A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash scripting language for the world wide web.</p>	<p>“The page server 110 is disposed for generating and responding to messages in a protocol for presenting web pages to the page client 130. The protocol is preferably the Hypertext Transfer Protocol (“HTTP”), but the concepts of the invention are broad enough to apply to other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols, such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 2, ll. 45-56.</p> <p>“In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 13-15.</p> <p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“Links 150 are embedded in pages 140 using a set of HTML tags, or by other techniques for identifying other pages 140 which may be presented. Links 150 are themselves presented to the operator as text elements 151 or as graphic elements 152. Links 150 may be identified by the page client 130 by special forms of display, for example by displaying text in a different color or typeface from text elements 151 or</p>

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		as graphic elements 152 which are not associated with links 150.” Col. 4, ll. 17-25. Nguyen ‘498 additionally cites several Java references, e.g., Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996; and Stephen R. Davis. “Teach Yourself Java Programming the Quick and Easy Way with Microsoft Visual: Learn Java Now”. Microsoft Press. 3 0402 00136 8093.
12	12. An information processing device comprising:	“A system 100 for accessing web pages comprises a page server 110, a communication link 120, and a page client 130.” Col. 2, ll. 38-39; <i>see also</i> FIG. 1.
12.1	a processor;	“The page client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, and program/data memory.” Col. 3, ll. 10-13.
12.2	a display;	“In a first preferred embodiment, the page client 130 comprises an input element 133 and a display element 134 . . . The display element 134 comprises a visual display element such as a monitor or a display panel[.]” Col. 3, ll. 22-27.
12.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; ;	A system 100 for accessing web pages comprises a page server 110, a communication link 120, and a page client 130.” Col. 2., ll. 37-39. “The page client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, and program/data memory.” Col. 3, ll. 10-13. “A primary page 140 presented on the output element 134 comprises information for presentation to an operator.” Col. 3, ll. 44-45.
12.4	wherein the hypermedia browser displays a temporary graphic element over the	“The page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
	content viewing area during times when the browser is loading visible content;	153 comprises a first part 154 having a first display color or pattern and a second part 153 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.
12.5	wherein the temporary graphic element is positioned only over a portion of the content viewing area and obstructs only part of the visible content in the content viewing area; and	In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 5, ll. 6-8; <i>see also</i> FIG. 1.
12.6	wherein the temporary graphic element indicates to a user that the browser is loading content and content comprises data for presentation which is from a source external to the browser.	“[T]he preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140. In this preferred embodiment, the first part 154 comprises a pie slice of the dot or small circle which by its angle and its color (green) indicates an amount of the “in progress” preloading stage which has been completed, and the second part 155 by its angle and its color (orange) indicates an amount which has not been completed. Thus the image of the dot or small circle will change color from orange to orange/green to green as the preloading operation progresses.” Col. 5, ll. 6-16. Content is provided by a page server external to the browser. <i>See</i> Col. 2, ll. 45-47; <i>see also</i> FIG. 1.
13	An information processing device as recited in claim 12, wherein the temporary graphic element is animated.	“In this preferred embodiment, the first part 154 comprises a pie slice of the dot or small circle which by its angle and its color (green) indicates an amount of the ‘in progress’ preloading stage which has been completed, and the second part 155 by its angle and its color (orange) indicates an amount which has not been completed. Thus the image of the dot or small circle will change color from orange to orange/green to

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device green as the preloading operation progresses.” Col. 5, ll. 8-16.
14	An information processing device as recited in claim 12, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	“In addition to the first part 154 and the second part 155, the preloading indicator 153 comprises a textual or graphical indicator, positioned at a margin of the primary page 140 (such as in a margin for a window used for presentation of 50 the visual elements of the primary page 140), indicating an amount of the “in progress” preloading state which has been completed. For example, when 75% of the secondary page 140 is preloaded, the preloading indicator 153 comprises the text “75% preloaded” or some equivalent statement, or 55 comprises a thermometer graph showing that 75% of the secondary page 140 is preloaded.” Col. 5, ll. 47-57; <i>see also</i> FIG. 1.
17	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The page server 110 is disposed for generating and responding to messages in a protocol for presenting web pages to the page client 130. The protocol is preferably the Hypertext Transfer Protocol (“HTTP”), but the concepts of the invention are broad enough to apply to other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols, such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 2, ll. 45-56.</p> <p>“In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 13-15.</p> <p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“Links 150 are embedded in pages 140 using a set of HTML tags, or by other</p>

	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
			techniques for identifying other pages 140 which may be presented. Links 150 are themselves presented to the operator as text elements 151 or as graphic elements 152. Links 150 may be identified by the page client 130 by special forms of display, for example by displaying text in a different color or typeface from text elements 151 or as graphic elements 152 which are not associated with links 150.” Col. 4, ll. 17-25. Nguyen ‘498 additionally cites several Java references, e.g., Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996; and Stephen R. Davis. “Teach Yourself Java Programming the Quick and Easy Way with Microsoft Visual: Learn Java Now”. Microsoft Press. 3 0402 00136 8093.
18	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.		<p>“The page server 110 is disposed for generating and responding to messages in a protocol for presenting web pages to the page client 130. The protocol is preferably the Hypertext Transfer Protocol (“HTTP”), but the concepts of the invention are broad enough to apply to other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols, such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 2, ll. 45-56.</p> <p>“In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 13-15.</p> <p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“Links 150 are embedded in pages 140 using a set of HTML tags, or by other techniques for identifying other pages 140 which may be presented. Links 150 are themselves presented to the operator as text elements 151 or as graphic elements 152.</p>

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		Links 150 may be identified by the page client 130 by special forms of display, for example by displaying text in a different color or typeface from text elements 151 or as graphic elements 152 which are not associated with links 150.” Col. 4, ll. 17-25. Nguyen ‘498 additionally cites several Java references, e.g., Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996; and Stephen R. Davis. “Teach Yourself Java Programming the Quick and Easy Way with Microsoft Visual: Learn Java Now”. Microsoft Press. 3 0402 00136 8093.
20	An information processing device as recited in claim 12, wherein the temporary graphic element is not content.	“[T]he preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140. In this preferred embodiment, the first part 154 comprises a pie slice of the dot or small circle which by its angle and its color (green) indicates an amount of the ‘in progress’ preloading stage which has been completed, and the second part 155 by its angle and its color (orange) indicates an amount which has not been completed. Thus the image of the dot or small circle will change color from orange to orange/green to green as the preloading operation progresses.” Col. 5, ll. 6-16.
21	An information processing device as recited in claim 12, wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.	“The choice of display colors is, of course, completely arbitrary. Other colors, patterns, or other visual or audible indicators may be used instead or in conjunction to present the operator with information about the pre loading stage for any particular link 150.” Col. 4, ll. 53-57. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete. Nguyen ‘498 discloses that the preloading indicator indicates an amount of the ‘in progress’ preloading stage which has been completed and an amount which has not been completed. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		<p>graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.</p> <p>In addition, the combination of Nguyen ‘498 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
32	A method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content, the method comprising:	“The page client 130 identifies each one of those links 150 as being in one of three preloading states [not started, in progress, or completed], using one of a set of display colors” Col. 4., ll. 38-40; see also TABLE 1-1.

	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device													
	<p style="text-align: center;">TABLE 1-1</p> <table> <tr> <th>Preloading State</th><th>Display Color</th><th>Nature of State</th></tr> <tr> <td>not started</td><td>red</td><td>the secondary page has not started preloading yet</td></tr> <tr> <td>in progress</td><td>orange</td><td>the secondary page is currently being preloaded</td></tr> <tr> <td>completed</td><td>green</td><td>the secondary page is preloaded and is ready for presentation</td></tr> </table> <p>“The page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator 153 comprises a first part 154 having a first display color or pattern and a second part 154 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.</p> <p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“The page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator 153 comprises a first part 154 having a first display color or pattern and a second part 154 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4,</p>	Preloading State	Display Color	Nature of State	not started	red	the secondary page has not started preloading yet	in progress	orange	the secondary page is currently being preloaded	completed	green	the secondary page is preloaded and is ready for presentation	
Preloading State	Display Color	Nature of State												
not started	red	the secondary page has not started preloading yet												
in progress	orange	the secondary page is currently being preloaded												
completed	green	the secondary page is preloaded and is ready for presentation												
32.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen being without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;													

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		<p>Il. 65-67 to Col. 5, Il. 1-8.</p> <p>“The choice of display colors is, of course, completely arbitrary. Other colors, patterns, or other visual or audible indicators may be used instead or in conjunction to present the operator with information about the preloading stage for any particular link 150.” Col. 4, Il. 53-57.</p>
32.2	receiving an instruction to load new content into the content viewing area;	<p>“The preloaded secondary pages 140 are stored in the client storage 132. When the operator follows one of the links 150 to one of the secondary pages 140, that secondary page 140 is presented to the operator by reference to the client storage 132, rather than by requesting that secondary page 140 from the page server 110.” Col. 5, Il. 58-63.</p> <p><i>See, e.g.</i>, Claim 1 (“loading a first page which has been dynamically selected by an operator”).</p>
32.3	loading such new content into the content viewing area; and	<p>“The preloaded secondary pages 140 are stored in the client storage 132. When the operator follows one of the links 150 to one of the secondary pages 140, that secondary page 140 is presented to the operator by reference to the client storage 132, rather than by requesting that secondary page 140 from the page server 110.” Col. 5, Il. 58-63.</p> <p>“When it is presented to the operator, the secondary page 140 logically becomes the primary page 140, preloading of any other secondary pages 140 is aborted, and the page client 130 preloads those (new) secondary pages 140 which are pointed to by the (new) primary page 140 in like manner as it preloaded those (old) secondary pages 140 which were pointed to by the (old) primary page 140. Dynamic preloading of secondary pages 140 continues so long as the operator continues to follow links 150 and thus request presentation of secondary pages 140, which logically become (new) primary pages 140.” Col. 6, Il. 36-46.</p>
32.4	while loading, displaying a “load status” graphic element over the content viewing	<p>“The page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator</p>

	U.S. Patent 6,339,780	area so that the graphic element obstructs only part of the content in such content viewing area; and	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		wherein content comprises data for presentation which is from a source external to the browser.	153 comprises a first part 154 having a first display color or pattern and a second part 153 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.
32.5		wherein content comprises data for presentation which is from a source external to the browser.	Content is provided by a page server external to the browser. See Col. 2, ll. 45-47; see also FIG. 1.
33	A method as recited in claim 32 further comprising, upon completion of the		“The choice of display colors is, of course, completely arbitrary. Other colors, patterns, or other visual or audible indicators may be used instead or in conjunction to

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
	loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	<p>present the operator with information about the pre loading stage for any particular link 150.” Col. 4, ll. 53-57.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete. Nguyen ‘498 discloses that the preloading indicator indicates an amount of the ‘in progress’ preloading stage which has been completed and an amount which has not been completed. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.</p> <p>In addition, the combination of Nguyen ‘498 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
34	A hypermedia browser of claim 32,	“The page server 110 is disposed for generating and responding to messages in a

	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
	wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.		<p>protocol for presenting web pages to the page client 130. The protocol is preferably the Hypertext Transfer Protocol (“HTTP”), but the concepts of the invention are broad enough to apply to other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols, such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 2, ll. 45-56.</p> <p>“In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 13-15.</p> <p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“Links 150 are embedded in pages 140 using a set of HTML tags, or by other techniques for identifying other pages 140 which may be presented. Links 150 are themselves presented to the operator as text elements 151 or as graphic elements 152. Links 150 may be identified by the page client 130 by special forms of display, for example by displaying text in a different color or typeface from text elements 151 or as graphic elements 152 which are not associated with links 150.” Col. 4, ll. 17-25.</p> <p>Nguyen ‘498 additionally cites several Java references, e.g., Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996; and Stephen R. Davis. “Teach Yourself Java Programming the Quick and Easy Way with Microsoft Visual: Learn Java Now”. Microsoft Press. 3 0402 00136 8093.</p>
35	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a		<p>“The page server 110 is disposed for generating and responding to messages in a protocol for presenting web pages to the page client 130. The protocol is preferably the Hypertext Transfer Protocol (“HTTP”), but the concepts of the invention are broad</p>

	U.S. Patent 6,339,780	<p>U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device</p> <p>enough to apply to other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols, such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 2, ll. 45-56.</p> <p>“In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 13-15.</p> <p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“Links 150 are embedded in pages 140 using a set of HTML tags, or by other techniques for identifying other pages 140 which may be presented. Links 150 are themselves presented to the operator as text elements 151 or as graphic elements 152. Links 150 may be identified by the page client 130 by special forms of display, for example by displaying text in a different color or typeface from text elements 151 or as graphic elements 152 which are not associated with links 150.” Col. 4, ll. 17-25.</p> <p>Nguyen ‘498 additionally cites several Java references, e.g., Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996; and Stephen R. Davis. “Teach Yourself Java Programming the Quick and Easy Way with Microsoft Visual: Learn Java Now”. Microsoft Press. 3 0402 00136 8093.</p>
36	A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method of indicating a content “load status” of a hypermedia browser having a	<p>“The page client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, and program/data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language. The general purpose processor may comprise any processor disposed to</p>

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
	content viewing area for viewing content, the method comprising:	<p>interpret or to compile the “Java” computer language, such as an Intel “Pentium” processor operating at 90 megahertz, having 32 megabytes of program/data memory, operating under control of the Microsoft “Windows 95” operating system, and coupled to 1.0 gigabytes of client storage 132.” Col. 3, ll. 10-21.</p> <p>“FIG. 1 shows a system for accessing web pages. A system 100 for accessing web pages comprises a page server 110, a communication link 120, and a page client 130.” Col. 2., ll. 36-39. The page client may be a web browser. Col. 1, ll. 52-53. “In a first preferred embodiment, the page client 130 comprises an input element 133 and a display element 134. The input element 133 comprises a keyboard and a pointing device such as a mouse or trackball. The display element 134 comprises a visual display element such as a monitor or a display panel[.]” Col. 3, ll. 22-27.</p> <p>“The page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator 153 comprises a first part 154 having a first display color or pattern and a second part 153 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.</p>
36.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen is without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;	<p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“The page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator 153 comprises a first part 154 having a first display color or pattern and a second part 153 having a second display color or pattern; the shape or size of the first part 154</p>

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8. “The choice of display colors is, of course, completely arbitrary. Other colors, patterns, or other visual or audible indicators may be used instead or in conjunction to present the operator with information about the preloading stage for any particular link 150.” Col. 4, ll. 53-57.
36.2	receiving an instruction to load new content into the content viewing area;	“The preloaded secondary pages 140 are stored in the client storage 132. When the operator follows one of the links 150 to one of the secondary pages 140, that secondary page 140 is presented to the operator by reference to the client storage 132, rather than by requesting that secondary page 140 from the page server 110.” Col. 5, ll. 58-63. <i>See, e.g.,</i> Claim 1 (“loading a first page which has been dynamically selected by an operator”).
36.3	loading such new content into the content viewing area; and	“The preloaded secondary pages 140 are stored in the client storage 132. When the operator follows one of the links 150 to one of the secondary pages 140, that secondary page 140 is presented to the operator by reference to the client storage 132, rather than by requesting that secondary page 140 from the page server 110.” Col. 5, ll. 58-63. “When it is presented to the operator, the secondary page 140 logically becomes the primary page 140, preloading of any other secondary pages 140 is aborted, and the page client 130 preloads those (new) secondary pages 140 which are pointed to by the (new) primary page 140 in like manner as it preloaded those (old) secondary pages 140 which were pointed to by the (old) primary page 140. Dynamic preloading of secondary pages 140 continues so long as the operator continues to follow links 150

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		and thus request presentation of secondary pages 140, which logically become (new) primary pages 140.” Col. 6, ll. 36-46.
36.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	“The page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator 153 comprises a first part 154 having a first display color or pattern and a second part 154 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.
36.5	wherein content comprises data for presentation which is from a source external to the browser.	Content is provided by a page server external to the browser. <i>See</i> Col. 2, ll. 45-47; <i>see also</i> FIG. 1.

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 ("Nguyen '498") alone and/or in combination with U.S. Patent 5,528,744 ("Vaughton '744") and/or U.S. Patent 6,377,978 ("Nguyen '978") and/or various systems implementing a web browser on a mobile device
		<p>The diagram illustrates a system architecture. On the left, a dashed box labeled "CLIENT" contains three components: a monitor (134), a central processing unit (131), and a keyboard (133). A database symbol (132) is connected to the CPU (131). Below the client box is a large rectangular area representing a page or document. This area contains several elements: a top-left section with two small boxes (161, 162); a middle section with horizontal lines (151, 152) and a cursor arrow; a bottom-left section with a box containing "75%"; a bottom-right section with a box containing "153"; and a right-side section with a box containing "154" and another box below it. To the right of the main page area are three separate boxes, each labeled "140". Dashed arrows indicate connections from the client's monitor (134) to the top-left section of the page, and from the client's CPU (131) to the three boxes labeled "140".</p>
37	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>"The page server 110 is disposed for generating and responding to messages in a protocol for presenting web pages to the page client 130. The protocol is preferably the Hypertext Transfer Protocol ("HTTP"), but the concepts of the invention are broad enough to apply to other protocols for transferring and presenting information, including protocols known as "FSP", "FTP", "Gopher", and variants thereof, protocols for access to a command interface such as "Telnet", "MUD", "MUSH", "MOO", and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols, such as "Archie", "Veronica", "Jughead", and the like." Col. 2, ll. 45-56.</p> <p>"In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the "Java" computer language." Col. 3, ll. 13-15.</p>

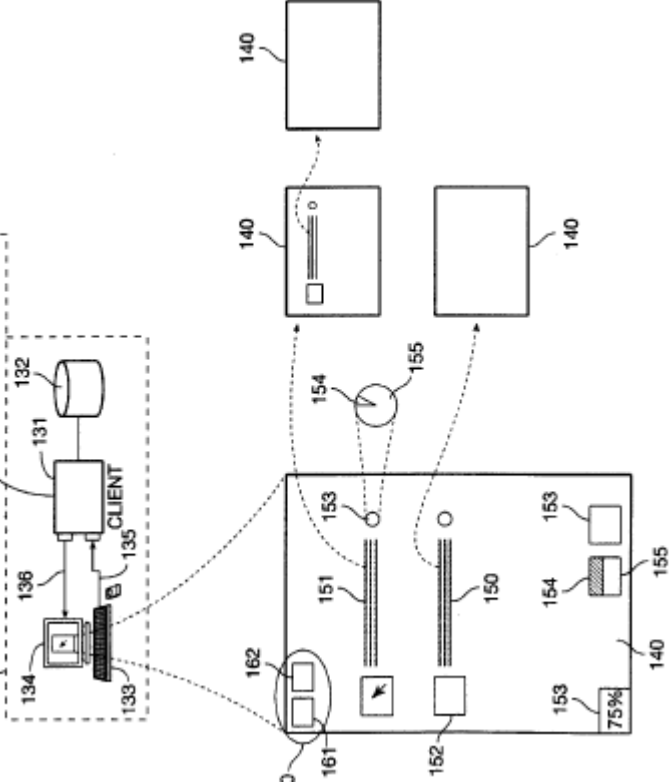
	U.S. Patent 6,339,780	<p>U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device</p> <p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“Links 150 are embedded in pages 140 using a set of HTML tags, or by other techniques for identifying other pages 140 which may be presented. Links 150 are themselves presented to the operator as text elements 151 or as graphic elements 152. Links 150 may be identified by the page client 130 by special forms of display, for example by displaying text in a different color or typeface from text elements 151 or as graphic elements 152 which are not associated with links 150.” Col. 4, ll. 17-25.</p> <p>Nguyen ‘498 additionally cites several Java references, e.g., Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996; and Stephen R. Davis. “Teach Yourself Java Programming the Quick and Easy Way with Microsoft Visual: Learn Java Now”. Microsoft Press. 3 0402 00136 8093.</p>
38	<p>A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.</p>	<p>“The page server 110 is disposed for generating and responding to messages in a protocol for presenting web pages to the page client 130. The protocol is preferably the Hypertext Transfer Protocol (“HTTP”), but the concepts of the invention are broad enough to apply to other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols, such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 2, ll. 45-56.</p> <p>“In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 13-15.</p> <p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form</p>

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50. “Links 150 are embedded in pages 140 using a set of HTML tags, or by other techniques for identifying other pages 140 which may be presented. Links 150 are themselves presented to the operator as text elements 151 or as graphic elements 152. Links 150 may be identified by the page client 130 by special forms of display, for example by displaying text in a different color or typeface from text elements 151 or as graphic elements 152 which are not associated with links 150.” Col. 4, ll. 17-25. Nguyen ‘498 additionally cites several Java references, e.g., Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996; and Stephen R. Davis. “Teach Yourself Java Programming the Quick and Easy Way with Microsoft Visual: Learn Java Now”. Microsoft Press. 3 0402 00136 8093.
39	A computer-readable medium as recited in claim 36 further having additional computer-executable instructions that perform a method comprising, upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	“The choice of display colors is, of course, completely arbitrary. Other colors, patterns, or other visual or audible indicators may be used instead or in conjunction to present the operator with information about the pre loading stage for any particular link 150.” Col. 4, ll. 53-57. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete. Nguyen ‘498 discloses that the preloading indicator indicates an amount of the ‘in progress’ preloading stage which has been completed and an amount which has not been completed. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete. In addition, the combination of Nguyen ‘498 and Vaughton ‘744 renders this claim

	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
			obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40	An information processing device comprising:		“A system 100 for accessing web pages comprises a page server 110, a communication link 120, and a page client 130.” Col. 2, ll. 38-39; <i>see also</i> FIG. 1.
40.1	a processor;		“The page client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, and program/data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language. The general purpose processor may comprise any processor disposed to interpret or to compile the “Java” computer language, such as an Intel “Pentium” processor operating at 90 megahertz, having 32 megabytes of program/data memory, operating under control of the Microsoft “Windows 95” operating system, and coupled to 1.0 gigabytes of client storage 132.” Col. 3, ll. 10-21.
40.2	a display;		“In a first preferred embodiment, the page client 130 comprises an input element 133

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		and a display element 134 . . . The display element 134 comprises a visual display element such as a monitor or a display panel[.]” Col. 3, ll. 22-27.
40.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser is configured to operate in a content-loading mode and a content-loaded mode;	<p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator.” Col. 3, ll. 44-45.</p> <p>“The page client 130 identifies each one of those links 150 as being in one of three preloading states [not started, in progress, or completed][.]” Col. 4., ll. 39-40; <i>see also</i> TABLE 1-1.</p>
40.4	in the content-loaded mode, the hypermedia browser displays loaded content in the content viewing area and no “load status” graphic element is displayed, wherein absence of such “load status” graphic element indicates that the browser is in the content-loaded mode;	<p>“The choice of display colors is, of course, completely arbitrary. Other colors, patterns, or other visual or audible indicators may be used instead or in conjunction to present the operator with information about the pre loading stage for any particular link 150.” Col. 4, ll. 53-57.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete. Nguyen ‘498 discloses that the preloading indicator indicates an amount of the ‘in progress’ preloading stage which has been completed and an amount which has not been completed. Based on this, one of skill in the art would have been motivated to implement a browser wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.</p> <p>In addition, the combination of Nguyen ‘498 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be</p>

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40.5	in the content-loading mode, the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a “load status” graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such “load status” graphic element indicates that the browser is in the content-loading mode; and	“The page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator 153 comprises a first part 154 having a first display color or pattern and a second part 154 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.
40.6	wherein content comprises data for presentation which is from a source external to the browser.	Content is provided by a page server external to the browser. <i>See</i> Col. 2, ll. 45-47; <i>see also</i> FIG. 1.

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		
41	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The page server 110 is disposed for generating and responding to messages in a protocol for presenting web pages to the page client 130. The protocol is preferably the Hypertext Transfer Protocol (“HTTP”), but the concepts of the invention are broad enough to apply to other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols, such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 2, ll. 45-56.</p> <p>“In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 13-15.</p>

	U.S. Patent 6,339,780	<p>U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device</p> <p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“Links 150 are embedded in pages 140 using a set of HTML tags, or by other techniques for identifying other pages 140 which may be presented. Links 150 are themselves presented to the operator as text elements 151 or as graphic elements 152. Links 150 may be identified by the page client 130 by special forms of display, for example by displaying text in a different color or typeface from text elements 151 or as graphic elements 152 which are not associated with links 150.” Col. 4, ll. 17-25.</p> <p>Nguyen ‘498 additionally cites several Java references, e.g., Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996; and Stephen R. Davis. “Teach Yourself Java Programming the Quick and Easy Way with Microsoft Visual: Learn Java Now”. Microsoft Press. 3 0402 00136 8093.</p>
42	<p>A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.</p>	<p>“The page server 110 is disposed for generating and responding to messages in a protocol for presenting web pages to the page client 130. The protocol is preferably the Hypertext Transfer Protocol (“HTTP”), but the concepts of the invention are broad enough to apply to other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols, such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 2, ll. 45-56.</p> <p>“In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 13-15.</p> <p>“A primary page 140 presented on the output element 134 comprises information for presentation to an operator. The information may be visual information (in the form</p>

	U.S. Patent 6,339,780	U.S. Patent 6,584,498 (“Nguyen ‘498”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,377,978 (“Nguyen ‘978”) and/or various systems implementing a web browser on a mobile device
		<p>of text, graphics, or motion picture data), may be audio information, or may be information in another format. There are several formats for information available via web pages which are known in the art of networking.” Col. 3, ll. 44-50.</p> <p>“Links 150 are embedded in pages 140 using a set of HTML tags, or by other techniques for identifying other pages 140 which may be presented. Links 150 are themselves presented to the operator as text elements 151 or as graphic elements 152. Links 150 may be identified by the page client 130 by special forms of display, for example by displaying text in a different color or typeface from text elements 151 or as graphic elements 152 which are not associated with links 150.” Col. 4, ll. 17-25.</p> <p>Nguyen ‘498 additionally cites several Java references, e.g., Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996; and Stephen R. Davis. “Teach Yourself Java Programming the Quick and Easy Way with Microsoft Visual: Learn Java Now”. Microsoft Press. 3 0402 00136 8093.</p>

EXHIBIT E

Exemplar Chart of U.S. Patent 6,339,780**U.S. Patent 6,377,978 (“Nguyen ‘978”)¹
Claims 1-6, 9-14, 17, 18, 20, 21 and 32-42**

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
1	A hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area,	<p>“The invention provides a method and system for dynamic downloading of hypertext electronic mail messages. The system includes a mail server for receiving electronic mail messages and their headers, and a mail client for downloading electronic mail messages and their headers from the mail receiver and presenting downloaded electronic mail messages and headers to an operator.” Col. 2, ll. 25-30.</p> <p>“In a first preferred embodiment, the mail client 130 comprises an input element 133 and a display element 134. The input element 133 comprises a keyboard and a pointing device such as a mouse or trackball. The display element 134 comprises a visual display element such as a monitor or a display panel[.]” Col. 3, ll. 44-50.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to access hypermedia email in a browser. For example, Nguyen ‘498, which is incorporated into Nguyen ‘978 discloses a web browser.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser on an information processing device having a limited display area.</p>

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Dynamic Loading of Hypertext Electronic Mail Messages” issued to Nguyen; filed September 13, 1996; issued April 23, 2002. In the specification, the patentee expressly incorporates his copending application (08/712,586), which later issued as U.S. Patent No. 6,584,498.

Claim	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device	U.S. Patent 6,339,780
	In addition, the combination of Nguyen ‘978 and one or more one or more references describing handheld devices with browsers renders this claim obvious. Gessler ² , Cooper ³ , Bartlett ⁴ , Watson ⁵ , Kamba ⁶ , and Lauff ⁷ each teach a hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area (e.g., a PDA, handheld, mobile device, etc.), and it would be obvious to one of skill in the art to modify Nguyen ‘978 with these teachings. ⁸ One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system). Additionally, Bartlett cites Gessler and Cooper cites Gessler and Bartlett. Moreover, all of these references are from the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.	
1.1	wherein the hypermedia browser has a	“As the mail client 130 downloads the electronic mail message 140, it presents as

² S. Gessler and A. Kotulla. PDAs as Mobile WWW Browsers. Computer Networks and ISDN Systems 28 (1995) 53-59 (“Gessler”) (see pgs. 53 and 56-58).

³ I. Cooper and R. Shufflebotham. PDWeb Browsers: Implementation Issues. (University of Kent at Canterbury, November 9, 1995) (“Cooper”) (see pgs. 3 and 5).

⁴ J. Bartlett. Experience with a Wireless World Wide Web Client. IEEE COMPCON 95 (San Francisco, March 1995) (“Bartlett”) (see pgs. 1, 4 and 5).

⁵ T. Watson. “Application Design for Wireless Computing” (1994 Mobile Computing Systems and Applications Workshop Position Paper (August 1994)).

⁶ T. Kamba, S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. (1996). “Using small screen space more efficiently.”

⁷ M. Lauff and W. Gellerson. Multimedia Client Implementation on Personal Digital Assistants. TecO.

⁸ Motorola continues to investigate other browsers that were in existence prior to the priority date of the ‘780 patent (e.g., NetHopper, Newt’s Cape, PocketWeb and Vagon).

Claim	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device	U.S. Patent 6,339,780
	content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing area during times when the browser is loading content,	much as possible of the electronic mail message 140 (such as a first page 143 of the electronic mail message 140) to the operator for dynamic review while the downloading operation is in progress.” Col. 5, l. 65 - Col. 6, l. 2. “The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1. Nguyen ‘498 discloses that “[i]n a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 5, ll. 6-8. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a temporary graphic element over the content viewing area during times when the browser is loading content. Placement of the temporary graphic element is a matter of design choice.
1.2	wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area,	“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1. Nguyen ‘498 discloses that “[t]he page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator 153 comprises a first part 154 having a first display color or pattern and a second part 153 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.
1.3	wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser.	“The progress indicator 151 may include a text element or a graphics element, having a first part 153 and a second part 154, each having different colors, and altering the relative sizes of the first part 151 and the second part 152 as the download operation progresses. In this embodiment, the graphics element comprises a separate dot or small circle 155 associated with the header 141; or, the text element comprises the

Claim	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device	<p>presentation of header 141 itself.</p> <p>The progress indicator 151 may include a text element 156 or a graphics element shown outside the header window 151. In this embodiment, the text element 156 comprises a phrase such as “75% complete” for a downloading operation which was in fact 75% complete; the graphics element may comprise a thermometer graph representing the progress of the downloading operation.” Col. 6, ll. 25-41.</p> <p>“The mail client 130 receives headers 141 from the mail server 110 and presents those headers 141 in the header window 151 to the operator.” Col. 5, ll. 21-23.</p> <p>“Responsive to the headers 141, the mail client 130 selects individual electronic mail messages 140 and transmits requests to the mail server 110 to download those individual electronic mail messages 140.” Col. 5, ll. 24-27.</p>
2	A hypermedia browser as recited in claim 1, wherein the browser is configured to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.	<p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>“The progress indicator 151 may include a text element or a graphics element, having a first part 153 and a second part 154, each having different colors, and altering the relative sizes of the first part 151 and the second part 152 as the download operation progresses. In this embodiment, the graphics element comprises a separate dot or small circle 155 associated with the header 141; or, the text element comprises the presentation of header 141 itself.</p> <p>The progress indicator 151 may include a text element 156 or a graphics element shown outside the header window 151. In this embodiment, the text element 156 comprises a phrase such as “75% complete” for a downloading operation which was in fact 75% complete; the graphics element may comprise a thermometer graph representing the progress of the downloading operation.” Col. 6, ll. 25-41.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such</p>

Claim	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device	<p>downloading of content is complete. Nguyen ‘978 discloses that mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages. Based on this, one of skill in the art would have been motivated to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete.</p> <p>In addition, the combination of Nguyen ‘978 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
3	A hypermedia browser as recited in claim 1, wherein the temporary graphic element indicates to a user that the browser is loading content.	“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.
4	A hypermedia browser as recited in claim 1, wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate	“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
	to a user that such loading of content is complete.	<p>“The progress indicator 151 may include a text element or a graphics element, having a first part 153 and a second part 154, each having different colors, and altering the relative sizes of the first part 151 and the second part 152 as the download operation progresses. In this embodiment, the graphics element comprises a separate dot or small circle 155 associated with the header 141; or, the text element comprises the presentation of header 141 itself.</p> <p>The progress indicator 151 may include a text element 156 or a graphics element shown outside the header window 151. In this embodiment, the text element 156 comprises a phrase such as “75% complete” for a downloading operation which was in fact 75% complete; the graphics element may comprise a thermometer graph representing the progress of the downloading operation.” Col. 6, ll. 25-41.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete. Nguyen ‘978 discloses that mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages. Based on this, one of skill in the art would have been motivated to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete.</p> <p>In addition, the combination of Nguyen ‘978 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area),</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
		disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
5	A hypermedia browser as recited in claim 1, wherein the temporary graphic element is animated.	“The progress indicator 151 may include a text element or a graphics element, having a first part 153 and a second part 154, each having different colors, and altering the relative sizes of the first part 151 and the second part 152 as the download operation progresses. In this embodiment, the graphics element comprises a separate dot or small circle 155 associated with the header 141; or, the text element comprises the presentation of header 141 itself.” Col. 6, ll. 25-33.
6	A hypermedia browser as recited in claim 1, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	See Fig. 1.
9	A hypermedia browser as recited in claim 1, wherein the temporary graphic element conveys status information of the browser.	“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.
10	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	“For example, it is often desirable to transmit information organized as hypertext, such as information presented using HTML (hypertext markup language) or related description languages, and capable of being transmitted using the HTTP (hypertext transfer protocol) or related protocols.” Col. 1, l. 66 - Col. 2, l. 4. “In preferred embodiments, the mail server and the mail client cooperate dynamically and interactively to download, so as to present to the operator, electronic mail messages, or portions thereof, linked by hypertext links and possibly including data, audiovisual material, included programs, security features, or other features in

Claim	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device	U.S. Patent 6,339,780
	<p>addition to text.” Col. 2, ll. 36-43.</p> <p>“The mail client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, program and data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 34-36.</p> <p>“In a preferred embodiment, each electronic mail message 140 may comprise links to actual web pages (i.e., web pages outside the collection of pages 143 comprising the electronic mail message 140 itself) stored at a web server or other server for accessing information. These actual web pages are accessed using the Hypertext Transfer Protocol (“HTTP”), or other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols., such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 4, ll. 25-39.</p> <p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein.” Col. 4, ll. 59-62.</p> <p>Nguyen ‘498 additionally cites the following Java reference: Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996.</p>	
11	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world	<p>“For example, it is often desirable to transmit information organized as hypertext, such as information presented using HTML (hypertext markup language) or related description languages, and capable of being transmitted using the HTTP (hypertext transfer protocol) or related protocols.” Col. 1, l. 66 - Col. 2, l. 4.</p> <p>“In preferred embodiments, the mail server and the mail client cooperate dynamically and interactively to download, so as to present to the operator, electronic mail messages, or portions thereof, linked by hypertext links and possibly including data,</p>

Claim	U.S. Patent 6,339,780	<p>U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device</p> <p>audiovisual material, included programs, security features, or other features in addition to text.” Col. 2, ll. 36-43.</p> <p>“The mail client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, program and data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 34-36.</p> <p>“In a preferred embodiment, each electronic mail message 140 may comprise links to actual web pages (i.e., web pages outside the collection of pages 143 comprising the electronic mail message 140 itself) stored at a web server or other server for accessing information. These actual web pages are accessed using the Hypertext Transfer Protocol (“HTTP”), or other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols., such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 4, ll. 25-39.</p> <p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein.” Col. 4, ll. 59-62.</p> <p>Nguyen ‘498 additionally cites the following Java reference: Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996.</p>
12	An information processing device comprising:	<p>“The invention provides a method and system for dynamic downloading of hypertext electronic mail messages. The system includes a mail server for receiving electronic mail messages and their headers, and a mail client for downloading electronic mail messages and their headers from the mail receiver and presenting downloaded electronic mail messages and headers to an operator.” Col. 2, ll. 25-31.</p> <p>“In the following description, a preferred embodiment of the invention is described</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
		with regard to preferred process steps and data structures. However, those skilled in the art would recognize, after perusal of this application, that embodiments of the invention may be implemented using a general purpose processor, and that modification of a general purpose processor to implement the process steps and data structures described herein would not require undue invention.” Col. 2, ll. 63-67.
12.1	a processor;	“In the following description, a preferred embodiment of the invention is described with regard to preferred process steps and data structures. However, those skilled in the art would recognize, after perusal of this application, that embodiments of the invention may be implemented using a general purpose processor, and that modification of a general purpose processor to implement the process steps and data structures described herein would not require undue invention.” Col. 2, ll. 63-67.
12.2	a display;	“In a first preferred embodiment, the mail client 130 comprises an input element 133 and a display element 134. The input element 133 comprises a keyboard and a pointing device such as a mouse or trackball. The display element 134 comprises a visual display element such as a monitor or a display panel, and an audio display element such as a speaker.” Col 3, ll. 44-50.
12.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser has a content viewing area for viewing content and wherein the hypermedia browser displays a temporary graphic element over the content viewing area during times when the browser is loading visible content;	<p>“As the mail client 130 downloads the electronic mail message 140, it presents as much as possible of the electronic mail message 140 (such as a first page 143 of the electronic mail message 140) to the operator for dynamic review while the downloading operation is in progress.” Col. 5, l. 65 - Col. 6, l. 2.</p> <p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>“When the operator selects a particular electronic mail message 140 for presentation, the mail client 130 determines if that particular electronic mail message 140 has been downloaded and is present in the client storage 132. If so, the mail client 130 presents that particular electronic mail message 140 from the client storage 132. If not, the mail client 130 transmits a request to the mail server 110 to download the newly selected electronic mail message 140, downloads the newly selected electronic mail</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
		<p>message 140, and presents the newly selected electronic mail message 140 to the operator in the page window 152.” Col. 6, ll. 41-52.</p> <p>Nguyen ‘498 discloses that “[i]n a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 5, ll. 6-8.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a temporary graphic element over the content viewing area during times when the browser is loading visible content. Placement of the temporary graphic element is a matter of design choice.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to access hypermedia email in a browser. For example, Nguyen ‘498, which is incorporated into Nguyen ‘978 discloses a web browser.</p>
12.4	wherein the temporary graphic element is positioned only over a portion of the content viewing area and obstructs only part of the visible content in the content viewing area; and	<p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>Nguyen ‘498 discloses that “[t]he page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator 153 comprises a first part 154 having a first display color or pattern and a second part 153 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.</p>
12.5	wherein the temporary graphic element indicates to a user that the browser is loading content and content comprises data for presentation which is from a source external to the browser.	<p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>“The progress indicator 151 may include a text element or a graphics element, having a first part 153 and a second part 154, each having different colors, and altering the</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
		relative sizes of the first part 151 and the second part 152 as the download operation progresses. In this embodiment, the graphics element comprises a separate dot or small circle 155 associated with the header 141; or, the text element comprises the presentation of header 141 itself. The progress indicator 151 may include a text element 156 or a graphics element shown outside the header window 151. In this embodiment, the text element 156 comprises a phrase such as “75% complete” for a downloading operation which was in fact 75% complete; the graphics element may comprise a thermometer graph representing the progress of the downloading operation.” Col. 6, ll. 25-41. “The mail client 130 receives headers 141 from the mail server 110 and presents those headers 141 in the header window 151 to the operator.” Col. 5, ll. 21-23. “Responsive to the headers 141, the mail client 130 selects individual electronic mail messages 140 and transmits requests to the mail server 110 to download those individual electronic mail messages 140.” Col. 5, ll. 24-27.
13	An information processing device as recited in claim 12, wherein the temporary graphic element is animated.	“The progress indicator 151 may include a text element or a graphics element, having a first part 153 and a second part 154, each having different colors, and altering the relative sizes of the first part 151 and the second part 152 as the download operation progresses. In this embodiment, the graphics element comprises a separate dot or small circle 155 associated with the header 141; or, the text element comprises the presentation of header 141 itself.” Col. 6, ll. 25-33.
14	An information processing device as recited in claim 12, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	See Fig. 1.
17	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a	“For example, it is often desirable to transmit information organized as hypertext, such as information presented using HTML (hypertext markup language) or related description languages, and capable of being transmitted using the HTTP (hypertext

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
	group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>transfer protocol) or related protocols.” Col. 1, l. 66 - Col. 2, l. 4.</p> <p>“In preferred embodiments, the mail server and the mail client cooperate dynamically and interactively to download, so as to present to the operator, electronic mail messages, or portions thereof, linked by hypertext links and possibly including data, audiovisual material, included programs, security features, or other features in addition to text.” Col. 2, ll. 36-43.</p> <p>“The mail client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, program and data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the ‘Java’ computer language.” Col. 3, ll. 34-36.</p> <p>“In a preferred embodiment, each electronic mail message 140 may comprise links to actual web pages (i.e., web pages outside the collection of pages 143 comprising the electronic mail message 140 itself) stored at a web server or other server for accessing information. These actual web pages are accessed using the Hypertext Transfer Protocol (“HTTP”), or other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols., such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 4, ll. 25-39.</p> <p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein.” Col. 4, ll. 59-62.</p> <p>Nguyen ‘498 additionally cites the following Java reference: Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996.</p>
18	A hypermedia browser of claim 12, wherein content is data formatted for	“For example, it is often desirable to transmit information organized as hypertext, such as information presented using HTML (hypertext markup language) or related

Claim	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device	U.S. Patent 6,339,780 presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.
	<p>description languages, and capable of being transmitted using the HTTP (hypertext transfer protocol) or related protocols.” Col. 1, l. 66 - Col. 2, l. 4.</p> <p>“In preferred embodiments, the mail server and the mail client cooperate dynamically and interactively to download, so as to present to the operator, electronic mail messages, or portions thereof, linked by hypertext links and possibly including data, audiovisual material, included programs, security features, or other features in addition to text.” Col. 2, ll. 36-43.</p> <p>“The mail client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, program and data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 34-36.</p> <p>“In a preferred embodiment, each electronic mail message 140 may comprise links to actual web pages (i.e., web pages outside the collection of pages 143 comprising the electronic mail message 140 itself) stored at a web server or other server for accessing information. These actual web pages are accessed using the Hypertext Transfer Protocol (“HTTP”), or other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols., such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 4, ll. 25-39.</p> <p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein.” Col. 4, ll. 59-62.</p> <p>Nguyen ‘498 additionally cites the following Java reference: Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRom Mar. 1996.</p>	
20	An information processing device as	“The progress indicator 151 may include a text element or a graphics element, having

Claim	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device	<p>U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device</p>
	recited in claim 12, wherein the temporary graphic element is not content.	<p>a first part 153 and a second part 154, each having different colors, and altering the relative sizes of the first part 151 and the second part 152 as the download operation progresses. In this embodiment, the graphics element comprises a separate dot or small circle 155 associated with the header 141; or, the text element comprises the presentation of header 141 itself.</p> <p>The progress indicator 151 may include a text element 156 or a graphics element shown outside the header window 151. In this embodiment, the text element 156 comprises a phrase such as “75% complete” for a downloading operation which was in fact 75% complete; the graphics element may comprise a thermometer graph representing the progress of the downloading operation.” Col. 6, ll. 25-41.</p> <p>“The mail client 130 receives headers 141 from the mail server 110 and presents those headers 141 in the header window 151 to the operator.” Col. 5, ll. 21-23.</p> <p>“Responsive to the headers 141, the mail client 130 selects individual electronic mail messages 140 and transmits requests to the mail server 110 to download those individual electronic mail messages 140.” Col. 5, ll. 24-27.</p>
21	An information processing device as recited in claim 12, wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.	<p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>“The progress indicator 151 may include a text element or a graphics element, having a first part 153 and a second part 154, each having different colors, and altering the relative sizes of the first part 151 and the second part 152 as the download operation progresses. In this embodiment, the graphics element comprises a separate dot or small circle 155 associated with the header 141; or, the text element comprises the presentation of header 141 itself.</p> <p>The progress indicator 151 may include a text element 156 or a graphics element shown outside the header window 151. In this embodiment, the text element 156 comprises a phrase such as “75% complete” for a downloading operation which was in fact 75% complete; the graphics element may comprise a thermometer graph</p>

Claim	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device representing the progress of the downloading operation.” Col. 6, ll. 25-41.
	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete. Nguyen ‘978 discloses that mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages. Based on this, one of skill in the art would have been motivated to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete.</p> <p>In addition, the combination of Nguyen ‘978 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
32	<p>A method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content, the method comprising:</p> <p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
		of skill in the art to access hypermedia email in a browser. For example, Nguyen ‘498, which is incorporated into Nguyen ‘978 discloses a web browser.
32.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen being without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;	<p>“As the mail client 130 downloads the electronic mail message 140, it presents as much as possible of the electronic mail message 140 (such as a first page 143 of the electronic mail message 140) to the operator for dynamic review while the downloading operation is in progress.” Col. 5, l. 65 - Col. 6, l. 2.</p> <p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p>
32.2	receiving an instruction to load new content into the content viewing area;	<p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein. In this aspect, the header window 151 is treated similarly to a web page in which each header 141 is treated similarly to a link to a first page 143 of its associated body 142.” Col. 4, ll. 59-65.</p> <p>“Thus, the operator may select one of the headers 141 so as to view its associated body 142, to which the mail client 130 responds by downloading that body 142 and presenting a first page 143 of that body 142.” Col. 4, l. 66 - col. 5, l. 2.</p>
32.3	loading such new content into the content viewing area; and	“Thus, the operator may select one of the headers 141 so as to view its associated body 142, to which the mail client 130 responds by downloading that body 142 and presenting a first page 143 of that body 142.” Col. 4, l. 66 - col. 5, l. 2.
32.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	<p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>Nguyen ‘498 discloses that “[t]he page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator 153 comprises a first part 154 having a first display color or pattern and a second part 153 having a second display color or pattern; the</p>

Claim	U.S. Patent 6,339,780	<p>U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device</p> <p>shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a “load status” graphic element over the content viewing area. Placement of the graphic element is a matter of design choice.</p>
32.5	wherein content comprises data for presentation which is from a source external to the browser.	<p>“The mail client 130 receives headers 141 from the mail server 110 and presents those headers 141 in the header window 151 to the operator.” Col. 5, ll. 21-23.</p> <p>“Responsive to the headers 141, the mail client 130 selects individual electronic mail messages 140 and transmits requests to the mail server 110 to download those individual electronic mail messages 140.” Col. 5, ll. 24-27.</p>
33	A method as recited in claim 32 further comprising, upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	<p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>“The progress indicator 151 may include a text element or a graphics element, having a first part 153 and a second part 154, each having different colors, and altering the relative sizes of the first part 151 and the second part 152 as the download operation progresses. In this embodiment, the graphics element comprises a separate dot or small circle 155 associated with the header 141; or, the text element comprises the presentation of header 141 itself.</p> <p>The progress indicator 151 may include a text element 156 or a graphics element shown outside the header window 151. In this embodiment, the text element 156 comprises a phrase such as “75% complete” for a downloading operation which was in fact 75% complete; the graphics element may comprise a thermometer graph representing the progress of the downloading operation.” Col. 6, ll. 25-41.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the progress indicator disappears</p>

Claim	U.S. Patent 6,339,780	<p>U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device</p> <p>when the downloading operation is complete to indicate to a user that such downloading of content is complete. Nguyen ‘978 discloses that mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages. Based on this, one of skill in the art would have been motivated to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete.</p> <p>In addition, the combination of Nguyen ‘978 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
34	<p>A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.</p>	<p>“For example, it is often desirable to transmit information organized as hypertext, such as information presented using HTML (hypertext markup language) or related description languages, and capable of being transmitted using the HTTP (hypertext transfer protocol) or related protocols.” Col. 1, l. 66 - Col. 2, l. 4.</p> <p>“In preferred embodiments, the mail server and the mail client cooperate dynamically and interactively to download, so as to present to the operator, electronic mail messages, or portions thereof, linked by hypertext links and possibly including data,</p>

Claim	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device	<p>audiovisual material, included programs, security features, or other features in addition to text.” Col. 2, ll. 36-43.</p> <p>“The mail client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, program and data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 34-36.</p> <p>“In a preferred embodiment, each electronic mail message 140 may comprise links to actual web pages (i.e., web pages outside the collection of pages 143 comprising the electronic mail message 140 itself) stored at a web server or other server for accessing information. These actual web pages are accessed using the Hypertext Transfer Protocol (“HTTP”), or other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols., such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 4, ll. 25-39.</p> <p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein.” Col. 4, ll. 59-62.</p> <p>Nguyen ‘498 additionally cites the following Java reference: Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDROM Mar. 1996.</p>
35	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X,	<p>“For example, it is often desirable to transmit information organized as hypertext, such as information presented using HTML (hypertext markup language) or related description languages, and capable of being transmitted using the HTTP (hypertext transfer protocol) or related protocols.” Col. 1, l. 66 - Col. 2, l. 4.</p> <p>“In preferred embodiments, the mail server and the mail client cooperate dynamically and interactively to download, so as to present to the operator, electronic mail</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
	Flash. scripting language for the world wide web.	<p>messages, or portions thereof, linked by hypertext links and possibly including data, audiovisual material, included programs, security features, or other features in addition to text.” Col. 2, ll. 36-43.</p> <p>“The mail client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, program and data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 34-36.</p> <p>“In a preferred embodiment, each electronic mail message 140 may comprise links to actual web pages (i.e., web pages outside the collection of pages 143 comprising the electronic mail message 140 itself) stored at a web server or other server for accessing information. These actual web pages are accessed using the Hypertext Transfer Protocol (“HTTP”), or other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols., such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 4, ll. 25-39.</p> <p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein.” Col. 4, ll. 59-62.</p> <p>Nguyen ‘498 additionally cites the following Java reference: Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996.</p>
36	A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method of indicating a content “load status” of a hypermedia browser having a	<p>“The invention provides a method and system for dynamic downloading of hypertext electronic mail messages. The system includes a mail server for receiving electronic mail messages and their headers, and a mail client for downloading electronic mail messages and their headers from the mail receiver and presenting downloaded electronic mail messages and headers to an operator.” Col. 2, ll. 25-31.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
	content viewing area for viewing content, the method comprising:	<p>“In the following description, a preferred embodiment of the invention is described with regard to preferred process steps and data structures. However, those skilled in the art would recognize, after perusal of this application, that embodiments of the invention may be implemented using a general purpose processor, and that modification of a general purpose processor to implement the process steps and data structures described herein would not require undue invention.” Col. 2, ll. 63-67.</p> <p>“As the mail client 130 downloads the electronic mail message 140, it presents as much as possible of the electronic mail message 140 (such as a first page 143 of the electronic mail message 140) to the operator for dynamic review while the downloading operation is in progress.” Col. 5, l. 65 - Col. 6, l. 2.</p> <p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to access hypermedia email in a browser. For example, Nguyen ‘498, which is incorporated into Nguyen ‘978 discloses a web browser.</p>
36.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen is without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;	<p>“As the mail client 130 downloads the electronic mail message 140, it presents as much as possible of the electronic mail message 140 (such as a first page 143 of the electronic mail message 140) to the operator for dynamic review while the downloading operation is in progress.” Col. 5, l. 65 - Col. 6, l. 2.</p> <p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p>
36.2	receiving an instruction to load new content into the content viewing area;	<p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein. In this aspect, the header window 151 is treated similarly to a web page in which each header 141 is treated similarly to a link to a first page 143 of its associated body 142.” Col. 4, ll. 59-65.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
		“Thus, the operator may select one of the headers 141 so as to view its associated body 142, to which the mail client 130 responds by downloading that body 142 and presenting a first page 143 of that body 142.” Col. 4, l. 66 - col. 5, l. 2.
36.3	loading such new content into the content viewing area; and	“Thus, the operator may select one of the headers 141 so as to view its associated body 142, to which the mail client 130 responds by downloading that body 142 and presenting a first page 143 of that body 142.” Col. 4, l. 66 - col. 5, l. 2.
36.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	<p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>Nguyen ‘498 discloses that “[t]he page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator 153 comprises a first part 154 having a first display color or pattern and a second part 153 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a “load status” graphic element over the content viewing area. Placement of the graphic element is a matter of design choice.</p>
36.5	wherein content comprises data for presentation which is from a source external to the browser.	<p>“The mail client 130 receives headers 141 from the mail server 110 and presents those headers 141 in the header window 151 to the operator.” Col. 5, ll. 21-23.</p> <p>“Responsive to the headers 141, the mail client 130 selects individual electronic mail messages 140 and transmits requests to the mail server 110 to download those individual electronic mail messages 140.” Col. 5, ll. 24-27.</p>
37	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a	“For example, it is often desirable to transmit information organized as hypertext, such as information presented using HTML (hypertext markup language) or related description languages, and capable of being transmitted using the HTTP (hypertext

Claim	U.S. Patent 6,339,780	<p>U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device</p> <p>transfer protocol) or related protocols.” Col. 1, l. 66 - Col. 2, l. 4.</p> <p>“In preferred embodiments, the mail server and the mail client cooperate dynamically and interactively to download, so as to present to the operator, electronic mail messages, or portions thereof, linked by hypertext links and possibly including data, audiovisual material, included programs, security features, or other features in addition to text.” Col. 2, ll. 36-43.</p> <p>“The mail client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, program and data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the ‘Java’ computer language.” Col. 3, ll. 34-36.</p> <p>“In a preferred embodiment, each electronic mail message 140 may comprise links to actual web pages (i.e., web pages outside the collection of pages 143 comprising the electronic mail message 140 itself) stored at a web server or other server for accessing information. These actual web pages are accessed using the Hypertext Transfer Protocol (“HTTP”), or other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols., such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 4, ll. 25-39.</p> <p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein.” Col. 4, ll. 59-62.</p> <p>Nguyen ‘498 additionally cites the following Java reference: Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996.</p>
38	A hypermedia browser of claim 36, wherein content is data formatted for	<p>“For example, it is often desirable to transmit information organized as hypertext, such as information presented using HTML (hypertext markup language) or related</p>

Claim	U.S. Patent 6,339,780	<p>U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device</p> <p>description languages, and capable of being transmitted using the HTTP (hypertext transfer protocol) or related protocols.” Col. 1, l. 66 - Col. 2, l. 4.</p> <p>“In preferred embodiments, the mail server and the mail client cooperate dynamically and interactively to download, so as to present to the operator, electronic mail messages, or portions thereof, linked by hypertext links and possibly including data, audiovisual material, included programs, security features, or other features in addition to text.” Col. 2, ll. 36-43.</p> <p>“The mail client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, program and data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 34-36.</p> <p>“In a preferred embodiment, each electronic mail message 140 may comprise links to actual web pages (i.e., web pages outside the collection of pages 143 comprising the electronic mail message 140 itself) stored at a web server or other server for accessing information. These actual web pages are accessed using the Hypertext Transfer Protocol (“HTTP”), or other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols., such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 4, ll. 25-39.</p> <p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein.” Col. 4, ll. 59-62.</p> <p>Nguyen ‘498 additionally cites the following Java reference: Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRROM Mar. 1996.</p>
39	A computer-readable medium as recited	“The mail client 130 presents the progress of downloading operations using a progress

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
	<p>in claim 36 further having additional computer-executable instructions that perform a method comprising, upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.</p>	<p>indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>“The progress indicator 151 may include a text element or a graphics element, having a first part 153 and a second part 154, each having different colors, and altering the relative sizes of the first part 151 and the second part 152 as the download operation progresses. In this embodiment, the graphics element comprises a separate dot or small circle 155 associated with the header 141; or, the text element comprises the presentation of header 141 itself.</p> <p>The progress indicator 151 may include a text element 156 or a graphics element shown outside the header window 151. In this embodiment, the text element 156 comprises a phrase such as “75% complete” for a downloading operation which was in fact 75% complete; the graphics element may comprise a thermometer graph representing the progress of the downloading operation.” Col. 6, ll. 25-41.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete. Nguyen ‘978 discloses that mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages. Based on this, one of skill in the art would have been motivated to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete.</p> <p>In addition, the combination of Nguyen ‘978 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
		another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40	An information processing device comprising:	<p>“The invention provides a method and system for dynamic downloading of hypertext electronic mail messages. The system includes a mail server for receiving electronic mail messages and their headers, and a mail client for downloading electronic mail messages and their headers from the mail receiver and presenting downloaded electronic mail messages and headers to an operator.” Col. 2, ll. 25-31.</p> <p>“In the following description, a preferred embodiment of the invention is described with regard to preferred process steps and data structures. However, those skilled in the art would recognize, after perusal of this application, that embodiments of the invention may be implemented using a general purpose processor, and that modification of a general purpose processor to implement the process steps and data structures described herein would not require undue invention.” Col. 2, ll. 63-67.</p>
40.1	a processor;	“In the following description, a preferred embodiment of the invention is described with regard to preferred process steps and data structures. However, those skilled in the art would recognize, after perusal of this application, that embodiments of the invention may be implemented using a general purpose processor, and that modification of a general purpose processor to implement the process steps and data structures described herein would not require undue invention.” Col. 2, ll. 63-67.
40.2	a display;	“In a first preferred embodiment, the mail client 130 comprises an input element 133 and a display element 134. The input element 133 comprises a keyboard and a pointing device such as a mouse or trackball. The display element 134 comprises a visual display element such as a monitor or a display panel, and an audio display

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device element such as a speaker.” Col 3, ll. 44-50.
40.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser is configured to operate in a content-loading mode and a content-loaded mode;	<p>“As the mail client 130 downloads the electronic mail message 140, it presents as much as possible of the electronic mail message 140 (such as a first page 143 of the electronic mail message 140) to the operator for dynamic review while the downloading operation is in progress.” Col. 5, l. 65 - Col. 6, l. 2.</p> <p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to access hypermedia email in a browser. For example, Nguyen ‘498, which is incorporated into Nguyen ‘978 discloses a web browser.</p>
40.4	in the content-loaded mode, the hypermedia browser displays loaded content in the content viewing area and no “load status” graphic element is displayed, wherein absence of such “load status” graphic element indicates that the browser is in the content-loaded mode;	<p>“As the mail client 130 downloads the electronic mail message 140, it presents as much as possible of the electronic mail message 140 (such as a first page 143 of the electronic mail message 140) to the operator for dynamic review while the downloading operation is in progress.” Col. 5, l. 65 - Col. 6, l. 2.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete. Nguyen ‘978 discloses that mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages. Based on this, one of skill in the art would have been motivated to implement a browser wherein the progress indicator disappears when the downloading operation is complete to indicate to a user that such downloading of content is complete.</p> <p>In addition, the combination of Nguyen ‘978 and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 in claim chart) and it would be</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
		obvious to one of skill in the art to modify Nguyen ‘498 with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system including a status indicator of the download progress), address the same problem (indicating the downloading progress within the content viewing area), disclose the same or similar techniques for superimposing a download status indicator over the content, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40.5	in the content-loading mode, the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a “load status” graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such “load status” graphic element indicates that the browser is in the content-loading mode; and	<p>“As the mail client 130 downloads the electronic mail message 140, it presents as much as possible of the electronic mail message 140 (such as a first page 143 of the electronic mail message 140) to the operator for dynamic review while the downloading operation is in progress.” Col. 5, l. 65 - Col. 6, l. 2.</p> <p>“The mail client 130 presents the progress of downloading operations using a progress indicator 151, in like manner as the page client presents the progress of preloading operations for web pages.” Col. 6, ll. 18-19. See Fig. 1.</p> <p>Nguyen ‘498 discloses that “[t]he page client 130 presents the preloading state using a preloading indicator 153. To indicate the progress of the “in progress” preloading stage, the preloading indicator 153 comprises a first part 154 having a first display color or pattern and a second part 153 having a second display color or pattern; the shape or size of the first part 154 relative to the second part 155 is used to indicate the progress of the “in progress” preloading stage. In a first preferred embodiment, the preloading indicator 153 comprises a dot or other shape which is superposed on the primary page 140.” Col. 4, ll. 65-67 to Col. 5, ll. 1-8.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a “load status” graphic element over the content viewing area. Placement of the graphic element is a matter of design choice.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
40.6	wherein content comprises data for presentation which is from a source external to the browser.	<p>“The mail client 130 receives headers 141 from the mail server 110 and presents those headers 141 in the header window 151 to the operator.” Col. 5, ll. 21-23.</p> <p>“Responsive to the headers 141, the mail client 130 selects individual electronic mail messages 140 and transmits requests to the mail server 110 to download those individual electronic mail messages 140.” Col. 5, ll. 24-27.</p>
41	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“For example, it is often desirable to transmit information organized as hypertext, such as information presented using HTML (hypertext markup language) or related description languages, and capable of being transmitted using the HTTP (hypertext transfer protocol) or related protocols.” Col. 1, l. 66 - Col. 2, l. 4.</p> <p>“In preferred embodiments, the mail server and the mail client cooperate dynamically and interactively to download, so as to present to the operator, electronic mail messages, or portions thereof, linked by hypertext links and possibly including data, audiovisual material, included programs, security features, or other features in addition to text.” Col. 2, ll. 36-43.</p> <p>“The mail client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, program and data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 34-36.</p> <p>“In a preferred embodiment, each electronic mail message 140 may comprise links to actual web pages (i.e., web pages outside the collection of pages 143 comprising the electronic mail message 140 itself) stored at a web server or other server for accessing information. These actual web pages are accessed using the Hypertext Transfer Protocol (“HTTP”), or other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols., such as “Archie”, “Veronica”, “Jughead”, and the like.” Col. 4, ll. 25-39.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
		<p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein.” Col. 4, ll. 59-62.</p> <p>Nguyen ‘498 additionally cites the following Java reference: Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996.</p>
42	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>“For example, it is often desirable to transmit information organized as hypertext, such as information presented using HTML (hypertext markup language) or related description languages, and capable of being transmitted using the HTTP (hypertext transfer protocol) or related protocols.” Col. 1, l. 66 - Col. 2, l. 4.</p> <p>“In preferred embodiments, the mail server and the mail client cooperate dynamically and interactively to download, so as to present to the operator, electronic mail messages, or portions thereof, linked by hypertext links and possibly including data, audiovisual material, included programs, security features, or other features in addition to text.” Col. 2, ll. 36-43.</p> <p>“The mail client 130 similarly comprises a processor 131 and client storage 132, with the processor 111 comprising a general purpose processor having a computing element, program and data memory. In a preferred embodiment, preferred process steps and data structures for the page client 130 are specified in the “Java” computer language.” Col. 3, ll. 34-36.</p> <p>“In a preferred embodiment, each electronic mail message 140 may comprise links to actual web pages (i.e., web pages outside the collection of pages 143 comprising the electronic mail message 140 itself) stored at a web server or other server for accessing information. These actual web pages are accessed using the Hypertext Transfer Protocol (“HTTP”), or other protocols for transferring and presenting information, including protocols known as “FSP”, “FTP”, “Gopher”, and variants thereof, protocols for access to a command interface such as “Telnet”, “MUD”, “MUSH”, “MOO”, and variants thereof, other protocols for accessing, transmitting, or presenting information, and programs making use of such protocols., such as</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 6,377,978 (“Nguyen ‘978”) alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 6,584,498 (“Nguyen ‘498”) and/or various systems implementing a web browser on a mobile device
		<p>“Archie”, “Veronica”, “Jughead”, and the like.” Col. 4, ll. 25-39.</p> <p>“It is one aspect of the invention that electronic mail messages 140 are presented to the operator in like manner as web pages are presented in the “Dynamic Preloading of Web Pages” co-pending application referred to herein.” Col. 4, ll. 59-62.</p> <p>Nguyen ‘498 additionally cites the following Java reference: Tessier, Tom, “Using Java Script to Create Interactive Web Pages”, Dr. Dobbs Journal on CDRUM Mar. 1996.</p>

EXHIBIT F

Exemplar Chart of U.S. Patent 6,339,780**Experience with a Wireless World Wide Web Client (“Bartlett”)¹
Claims 1-6, 9-14, 17, 18, 20, 21 and 32-42**

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
1	1. A hypermedia browser embodied on a computer-readable medium for execution on an information processing device having a limited display area,	<p>“In order to separate promises from practice in PDAs and wireless communications, we decided to try to build a wireless PDA-based client to access the World Wide Web. Using equipment available in the winter of 1994, we were able to build a client that we call W4, the Wireless World Wide Web.” Abstract.</p> <p>“Today’s PDAs, with their limited computational power, storage, wireless communication bandwidth, and display size, offer a challenge: can you build anything, and if so, would anyone want to use it?” pg. 1.</p> <p>“Sections 2 through 5 provide a roughly chronological record of the design and implementation of W4, a ‘proof of concept’ for a Wireless World Wide Web client.” pg. 1.</p> <p><i>See also</i> Bartlett’s reference to Gessler²’s implementation of a browser on a mobile device. The analysis in this chart similarly applies to the mobile browsers described by Gessler, Cooper³, Watson⁴, Kamba⁵, and Lauff^{6,7}.</p>

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ J. Bartlett. Experience with a Wireless World Wide Web Client. IEEE COMPCON 95 (San Francisco, March 1995) (“Bartlett”).

² S. Gessler and A. Kotulla. PDAs as Mobile WWW Browsers. Computer Networks and ISDN Systems 28 (1995) 53-59 (“Gessler”).

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
1.1	wherein the hypermedia browser has a content viewing area for viewing content and is configured to display a temporary graphic element over the content viewing area during times when the browser is loading content,	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document. Each screen is identified with a tag that contains the document’s Universal Resource Location (URL) and an offset within the document. In the simplest PDA-based client, when a hypertext link or scroll arrow is tapped by the user, the tag associated with the interactor is sent in a request to the workstation. The workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replays to the PDA’s request.” pg. 3.</p> <p>“A pleasant discovery is that most Web documents are quite readable on a 320x240 pixel screen, even though they were designed for a much larger display.” pg. 4.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a temporary graphic element over the content viewing area during times when the browser is loading content. For example, according to Bartlett, “the workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replays to the PDA’s request.” Thus one of skill in the art would have been motivated to display a</p>

(...Continued)

³ I. Cooper and R. Shufflebotham. PDWeb Browsers: Implementation Issues. (University of Kent at Canterbury, November 9, 1995) (“Cooper”) (see pgs. 3 and 5).

⁴ T. Watson. “Application Design for Wireless Computing” (1994 Mobile Computing Systems and Applications Workshop Position Paper (August 1994)).

⁵ T. Kamba, S. A. Elson, T. Harpold, T. Stamper, and P. Sukaviriya. (1996). “Using small screen space more efficiently.”

⁶ M. Lauff and W. Gellerson. Multimedia Client Implementation on Personal Digital Assistants. TecO.

⁷ Motorola continues to investigate other browsers that were in existence prior to the priority date of the ‘780 patent (e.g., NetHopper, Newt’s Cape, PocketWeb and Vagon).

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		<p>temporary graphic element over the content viewing area during times when the browser is loading content.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches displaying a temporary graphic element over the content viewing area during times information is being downloaded (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are each in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.2	wherein the temporary graphic element is positioned over the content viewing area to obstruct only part of the content in the content viewing area,	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to position the temporary graphic element over the content viewing area to obstruct only part of the content in the content viewing area.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.3	wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the temporary graphic element is not content and wherein content comprises data for presentation which is from a source external to the browser. Bartlett notes that “[t]he workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replies to the PDA’s request.” pg. 3.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches a temporary graphic element that is not content (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
2	A hypermedia browser as recited in claim 1, wherein the browser is configured to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display the temporary graphic element over the content viewing area only during times when the browser is loading visible content.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches displaying the temporary graphic element over the content viewing area only during times when the browser is loading visible content (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
3	A hypermedia browser as recited in claim 1, wherein the temporary graphic element indicates to a user that the browser is loading content.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a temporary graphic element indicating to a user that the browser is loading content. In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches a temporary graphic element indicating to a user that the browser is loading content (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
4	A hypermedia browser as recited in claim 1, wherein the temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete. In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches a temporary graphic element that disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
5	A hypermedia browser as recited in claim 1, wherein the temporary graphic element is animated.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to animate the temporary graphic element. In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches a animating the temporary graphic element (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
6	A hypermedia browser as recited in claim 1, wherein the hypermedia browser displays the temporary graphic element	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area. Displaying the

	U.S. Patent 6,339,780	<p>Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device</p>
	in a corner of the content viewing area.	<p>temporary graphic element in a corner of the content viewing area is an obvious design choice to minimize the amount of content obstructed and would have been obvious to try.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches displaying the temporary graphic element in a corner of the content viewing area (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
9	A hypermedia browser as recited in claim 1, wherein the temporary graphic element conveys status information of the browser.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convey status information of the browser with a temporary graphic element.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches conveying status information of the browser with a temporary graphic element (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
10	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Bartlett with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language. Bartlett discloses a system that implements a World Wide Web browser on a mobile device.</p> <p>See also Gessler at pg. 57, cited by Bartlett at pg. 6 (“In our current version we support the most common structures of HTML and HTML + and images. The next release will include a complete HTML + interpreter which will also allow sensitive maps or tables on a PDA.”)</p>
11	A hypermedia browser of claim 1, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Bartlett with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web. Bartlett discloses a system that implements a World Wide Web browser on a mobile device.</p> <p>See also Gessler at pg. 57, cited by Bartlett at pg. 6 (“In our current version we support the most common structures of HTML and HTML + and images. The next release will include a complete HTML + interpreter which will also allow sensitive</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		maps or tables on a PDA.”)
12	12. An information processing device comprising:	Bartlett was implemented on an Apple MessagePad (i.e., Newton).
12.1	a processor;	“When designing for PDAs it might be dangerous to assume that processor speeds will rapidly improve. Cost, size, and power reduction seem to be more important than performance improvements. When more processor power is provided, it may not be available to the application programmer as it could be used to provide such things as software modems, improved handwriting recognition, or animation.” pg. 4.
12.2	a display;	“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.
12.3	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser has a content viewing area for viewing content and wherein the hypermedia browser displays a temporary graphic element over the content viewing area during times when the browser is loading visible content;	<p>“In order to separate promises from practice in PDAs and wireless communications, we decided to try to build a wireless PDA-based client to access the World Wide Web. Using equipment available in the winter of 1994, we were able to build a client that we call W4, the Wireless World Wide Web.” Abstract.</p> <p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document. Each screen is identified with a tag that contains the document’s Universal Resource Location (URL) and an offset within the document. In the simplest PDA-based client, when a hypertext link or scroll arrow is tapped by the user, the tag associated with the interactor is sent in a request to the workstation. The workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replies to the PDA’s request.” pg. 3.</p> <p>“A pleasant discovery is that most Web documents are quite readable on a 320x240 pixel screen, even though they were designed for a much larger display.” pg. 4.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a temporary graphic element over the content viewing area during times when the browser is loading content. For example, according to Bartlett, “the workstation obtains the document from the Web, parses it (caching the result for</p>

	U.S. Patent 6,339,780	<p>Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device</p>
		<p>later requests), formats the desired screen for the PDA, and then replays to the PDA’s request.” Thus one of skill in the art would have been motivated to display a temporary graphic element over the content viewing area during times when the browser is loading content.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches displaying a temporary graphic element over the content viewing area during times information is being downloaded (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are each in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
12.4	wherein the temporary graphic element is positioned only over a portion of the content viewing area and obstructs only part of the visible content in the content viewing area; and	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to position the temporary graphic element over the content viewing area to obstruct only part of the content in the content viewing area.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches a temporary graphic element positioned over the content viewing area to obstruct only part of the content in the content viewing area (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
12.5	wherein the temporary graphic element indicates to a user that the browser is loading content and content comprises data for presentation which is from a source external to the browser.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a temporary graphic element indicating to a user that the browser is loading content.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches a temporary graphic element indicating to a user that the browser is loading content (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein content comprises data for presentation which is from a source external to the browser. Bartlett notes that “[t]he workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replays to the PDA’s request.” pg. 3.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teach content comprising data for presentation which is from a source external to the browser. (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
13	An information processing device as recited in claim 12, wherein the temporary graphic element is animated.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to animate the temporary graphic element. In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches a animating the temporary graphic element (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
14	An information processing device as recited in claim 12, wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser displays the temporary graphic element in a corner of the content viewing area. Displaying the temporary graphic element in a corner of the content viewing area is an obvious design choice to minimize the amount of content obstructed and would have been obvious to try.

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		In addition, the combination of Bartlett and Vaughton ‘744 renders this claim obvious. Vaughton ‘744 teaches displaying the temporary graphic element in a corner of the content viewing area (see analysis of Vaughton ‘744 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
17	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Bartlett with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language. Bartlett discloses a system that implements a World Wide Web browser on a mobile device.</p> <p>See also Gessler at pg. 57, cited by Bartlett at pg. 6 (“In our current version we support the most common structures of HTML and HTML + and images. The next release will include a complete HTML + interpreter which will also allow sensitive maps or tables on a PDA.”)</p>
18	A hypermedia browser of claim 12, wherein content is data formatted for presentation which is selected from a	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the</p>

	U.S. Patent 6,339,780	<p>Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device</p> <p>function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Bartlett with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web. Bartlett discloses a system that implements a World Wide Web browser on a mobile device.</p> <p>See also Gessler at pg. 57, cited by Bartlett at pg. 6 (“In our current version we support the most common structures of HTML and HTML + and images. The next release will include a complete HTML + interpreter which will also allow sensitive maps or tables on a PDA.”)</p>
20	An information processing device as recited in claim 12, wherein the temporary graphic element is not content.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the temporary graphic element is not content.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches a temporary graphic element that is not content (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
21	An information processing device as recited in claim 12, wherein the	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to have the temporary graphic element disappear when the browser’s</p>

	U.S. Patent 6,339,780	<p>Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device</p>
	<p>temporary graphic element disappears when the browser’s loading of content is complete to indicate to a user that such loading of content is complete.</p>	<p>loading of content is complete to indicate to a user that such loading of content is complete.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches having the temporary graphic element disappear when the browser’s loading of content is complete to indicate to a user that such loading of content is complete. (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
32	<p>32. A method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content, the method comprising:</p>	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to indicate a content “load status” of a hypermedia browser having a content viewing area for viewing content.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
32.1	displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen being without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document. Each screen is identified with a tag that contains the document’s Universal Resource Location (URL) and an offset within the document. In the simplest PDA-based client, when a hypertext link or scroll arrow is tapped by the user, the tag associated with the interactor is sent in a request to the workstation. The workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replies to the PDA’s request.” pg. 3.</p> <p>“A pleasant discovery is that most Web documents are quite readable on a 320x240 pixel screen, even though they were designed for a much larger display.” pg. 4.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art for the “load status” graphic element to indicate a current content load status of the hypermedia browser.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches a “load status” graphic element indicating a current content load status of the hypermedia browser (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
32.2	receiving an instruction to load new	“The PDA acts as a video-text client, displaying screens representing a portion of a

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
	content into the content viewing area;	hypertext document. Each screen is identified with a tag that contains the document’s Universal Resource Location (URL) and an offset within the document. In the simplest PDA-based client, when a hypertext link or scroll arrow is tapped by the user, the tag associated with the interactor is sent in a request to the workstation. The workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replays to the PDA’s request.” pg. 3.
32.3	loading such new content into the content viewing area; and	“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document. Each screen is identified with a tag that contains the document’s Universal Resource Location (URL) and an offset within the document. In the simplest PDA-based client, when a hypertext link or scroll arrow is tapped by the user, the tag associated with the interactor is sent in a request to the workstation. The workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replays to the PDA’s request.” pg. 3.
32.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area. In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
32.5	wherein content comprises data for presentation which is from a source external to the browser.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein content comprises data for presentation which is from a source external to the browser. Bartlett notes that “[t]he workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replies to the PDA’s request.” pg. 3.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teach content comprising data for presentation which is from a source external to the browser. (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
33	A method as recited in claim 32 further comprising, upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to upon completion of the loading, remove the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches upon completion of the loading, removing the “load status”</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
34	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Bartlett with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language. Bartlett discloses a system that implements a World Wide Web browser on a mobile device.</p> <p>See also Gessler at pg. 57, cited by Bartlett at pg. 6 (“In our current version we support the most common structures of HTML and HTML + and images. The next release will include a complete HTML + interpreter which will also allow sensitive maps or tables on a PDA.”)</p>
35	A hypermedia browser of claim 32, wherein content is data formatted for presentation which is selected from a	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
	group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Bartlett with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web. Bartlett discloses a system that implements a World Wide Web browser on a mobile device. See also Gessler at pg. 57, cited by Bartlett at pg. 6 (“In our current version we support the most common structures of HTML and HTML + and images. The next release will include a complete HTML + interpreter which will also allow sensitive maps or tables on a PDA.”)
36	A computer-readable medium having computer-executable instructions that, when executed by a computer, perform a method of indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content, the method comprising:	<p>“When designing for PDAs it might be dangerous to assume that processor speeds will rapidly improve. Cost, size, and power reduction seem to be more important than performance improvements. When more processor power is provided, it may not be available to the application programmer as it could be used to provide such things as software modems, improved handwriting recognition, or animation.” pg. 4.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to indicate a content “load status” of a hypermedia browser having a content viewing area for viewing content.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches indicating a content “load status” of a hypermedia browser having a content viewing area for viewing content (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to</p>

	U.S. Patent 6,339,780	<p>Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device</p>
		<p>combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
36.1	<p>displaying loaded content within the content viewing area of a screen of a hypermedia browser, the screen is without a “load status” graphic element, wherein a “load status” graphic element indicates a current content load status of the hypermedia browser;</p>	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document. Each screen is identified with a tag that contains the document’s Universal Resource Location (URL) and an offset within the document. In the simplest PDA-based client, when a hypertext link or scroll arrow is tapped by the user, the tag associated with the interactor is sent in a request to the workstation. The workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replays to the PDA’s request.” pg. 3.</p> <p>“A pleasant discovery is that most Web documents are quite readable on a 320x240 pixel screen, even though they were designed for a much larger display.” pg. 4.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art for the “load status” graphic element to indicate a current content load status of the hypermedia browser.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches a “load status” graphic element indicating a current content load status of the hypermedia browser (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
36.2	receiving an instruction to load new content into the content viewing area;	“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document. Each screen is identified with a tag that contains the document’s Universal Resource Location (URL) and an offset within the document. In the simplest PDA-based client, when a hypertext link or scroll arrow is tapped by the user, the tag associated with the interactor is sent in a request to the workstation. The workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replies to the PDA’s request.” pg. 3.
36.3	loading such new content into the content viewing area; and	“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document. Each screen is identified with a tag that contains the document’s Universal Resource Location (URL) and an offset within the document. In the simplest PDA-based client, when a hypertext link or scroll arrow is tapped by the user, the tag associated with the interactor is sent in a request to the workstation. The workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replies to the PDA’s request.” pg. 3.
36.4	while loading, displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area; and	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to display a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area. In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teaches displaying a “load status” graphic element over the content viewing area so that the graphic element obstructs only part of the content in such content viewing area (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
36.5	wherein content comprises data for presentation which is from a source external to the browser.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein content comprises data for presentation which is from a source external to the browser. Bartlett notes that “[t]he workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replays to the PDA’s request.” pg. 3.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teach content comprising data for presentation which is from a source external to the browser. (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
37	A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language.	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Bartlett with a hypermedia</p>

	U.S. Patent 6,339,780	<p>Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device</p>
		<p>browser wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup language, and visible results of a scripting language. Bartlett discloses a system that implements a World Wide Web browser on a mobile device.</p> <p>See also Gessler at pg. 57, cited by Bartlett at pg. 6 (“In our current version we support the most common structures of HTML and HTML + and images. The next release will include a complete HTML + interpreter which will also allow sensitive maps or tables on a PDA.”)</p>
38	<p>A hypermedia browser of claim 36, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.</p>	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Bartlett with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web. Bartlett discloses a system that implements a World Wide Web browser on a mobile device.</p> <p>See also Gessler at pg. 57, cited by Bartlett at pg. 6 (“In our current version we support the most common structures of HTML and HTML + and images. The next release will include a complete HTML + interpreter which will also allow sensitive maps or tables on a PDA.”)</p>
39	<p>A computer-readable medium as recited in claim 36 further having additional computer-executable instructions that perform a method comprising, upon completion of the loading, removing the “load status” graphic element to reveal</p>	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to upon completion of the loading, remove the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
	the part of the content in the content viewing area that the graphic element obstructed when the element was displayed.	Lavey ‘698 teaches upon completion of the loading, removing the “load status” graphic element to reveal the part of the content in the content viewing area that the graphic element obstructed when the element was displayed (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40	40. An information processing device comprising:	Bartlett was implemented on an Apple MessagePad (i.e., Newton).
40.1	a processor;	“When designing for PDAs it might be dangerous to assume that processor speeds will rapidly improve. Cost, size, and power reduction seem to be more important than performance improvements. When more processor power is provided, it may not be available to the application programmer as it could be used to provide such things as software modems, improved handwriting recognition, or animation.” pg. 4.
40.2	a display;	“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.
40.2	a hypermedia browser executing on the processor to load and display content in a content viewing area on the display; wherein the hypermedia browser is configured to operate in a content-loading mode and a content-loaded mode;	“In order to separate promises from practice in PDAs and wireless communications, we decided to try to build a wireless PDA-based client to access the World Wide Web. Using equipment available in the winter of 1994, we were able to build a client that we call W4, the Wireless World Wide Web.” Abstract. “The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document. Each screen is identified with a tag that contains the document’s Universal Resource Location (URL) and an offset within the document. In the

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		<p>simplest PDA-based client, when a hypertext link or scroll arrow is tapped by the user, the tag associated with the interactor is sent in a request to the workstation. The workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replays to the PDA’s request.” pg. 3.</p> <p>“A pleasant discovery is that most Web documents are quite readable on a 320x240 pixel screen, even though they were designed for a much larger display.” pg. 4.</p> <p>To the extent not expressly or inherently disclosed, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teach a hypermedia browser configured to operate in a content-loading mode and a content-loaded mode (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are each in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
40.3	in the content-loaded mode, the hypermedia browser displays loaded content in the content viewing area and no “load status” graphic element is displayed, wherein absence of such “load status” graphic element indicates that the browser is in the content-loaded mode;	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser displays loaded content in the content viewing area and no “load status” graphic element is displayed, wherein absence of such “load status” graphic element indicates that the browser is in the content-loaded mode.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teach the hypermedia browser displaying loaded content in the content viewing area and no “load status” graphic element is displayed, wherein absence of</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		such “load status” graphic element indicates that the browser is in the content-loaded mode (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40.4	in the content-loading mode, the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a “load status” graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such “load status” graphic element indicates that the browser is in the content-loading mode;	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a “load status” graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such “load status” graphic element indicates that the browser is in the content-loading mode.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teach the hypermedia browser loads content, displays such content in the content viewing area as it loads, and displays a “load status” graphic element over the content view area obstructing part of the content displayed in the content viewing area, wherein presence of such “load status” graphic element indicates that the browser is in the content-loading mode (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
40.5	wherein content comprises data for presentation which is from a source external to the browser.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement a browser wherein content comprises data for presentation which is from a source external to the browser. Bartlett notes that “[t]he workstation obtains the document from the Web, parses it (caching the result for later requests), formats the desired screen for the PDA, and then replies to the PDA’s request.” pg. 3.</p> <p>In addition, the combination of Bartlett and Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 renders this claim obvious. Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 teach content comprising data for presentation which is from a source external to the browser. (see analysis of Vaughton ‘744 and/or Judson ‘643 and/or Lavey ‘698 in claim chart) and it would be obvious to one of skill in the art to modify Bartlett with these teachings. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (systems for downloading files from another system), address the same problem, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
41	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a scripting language, and visible results of a	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Bartlett with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of visible effects of a markup language, visible text of such a markup</p>

	U.S. Patent 6,339,780	Bartlett alone and/or in combination with U.S. Patent 5,528,744 (“Vaughton ‘744”) and/or U.S. Patent 5,572,643 (“Judson ‘643”) and/or Lavey U.S. Patent 6,023,698 (“Lavey ‘698”) and/or various systems implementing a web browser on a mobile device
		language, and visible results of a scripting language. Bartlett discloses a system that implements a World Wide Web browser on a mobile device. See also Gessler at pg. 57, cited by Bartlett at pg. 6 (“In our current version we support the most common structures of HTML and HTML + and images. The next release will include a complete HTML + interpreter which will also allow sensitive maps or tables on a PDA.”)
42	A hypermedia browser of claim 40, wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web.	<p>“The PDA acts as a video-text client, displaying screens representing a portion of a hypertext document.” pg. 3.</p> <p>One of ordinary skill in the art would recognize that a hypermedia browser has the function of downloading files from another system when, for example, a user selects a hypermedia link. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement Bartlett with a hypermedia browser wherein content is data formatted for presentation which is selected from a group consisting of HTML, text, SGML, XML, java, XHTML, JavaScript, streaming video, VRML, Active X, Flash. scripting language for the world wide web. Bartlett discloses a system that implements a World Wide Web browser on a mobile device.</p> <p>See also Gessler at pg. 57, cited by Bartlett at pg. 6 (“In our current version we support the most common structures of HTML and HTML + and images. The next release will include a complete HTML + interpreter which will also allow sensitive maps or tables on a PDA.”)</p>

EXHIBIT G

Exemplar Chart of U.S. Patent 7,411,582**U.S. Patent 5,638,501 (“Gough ‘501’”)¹
Claims 1-4, 6, 8-11, 13-23, 25-31**

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt ² and/or U.S. Patent No. 5,603,053 ³ and/or the MessagePad 120 ⁴
1	1. In a computing environment, a computer-implemented method comprising:	“This invention relates generally to computer systems, and more particularly to graphical user interfaces for computer systems.” Col. 1, ll. 5-7.
1.1	displaying an actuatable icon representative of an input method	The invention provides for multiple types of overlays to be used as inputs. Col. 7, ll. 60-63 (“Of course, other overlay images besides keyboards can be provided by the

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

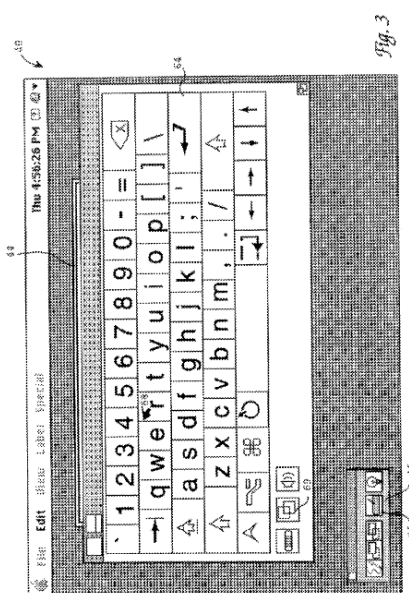
¹ “Method and Apparatus for Displaying an Overlay Image” issued to Gough et al.; filed May 10, 1993; issued June 10, 1997.

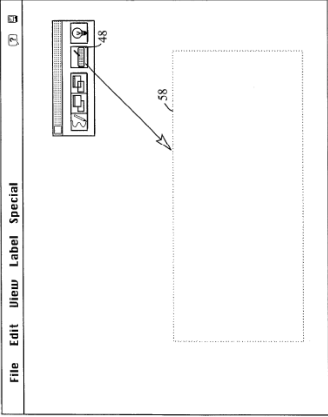
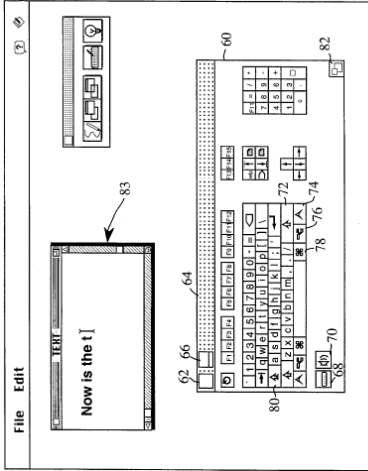
² Kraig Brockschmidt, Inside OLE (2d ed. 1995).

³ The Gough ‘501 patent incorporates by reference copending U.S. patent application Serial No. 08/060,438 (mistakenly listed as 08/060,458) entitled “Method and Apparatus for Interfacing With a Computer System.” Its inventors abandoned the ‘438 application in favor of a child application, U.S. patent application Serial No. 08/610,852, which they filed on March 5, 1996. The ‘852 application issued as U.S. Patent No. 5,603,053 (“Gough ‘053”) on February 11, 1997. The Gough ‘053 patent issued to Gough et al. and is entitled “System For Entering Data Into An Active Application Currently Running In The Foreground By Selecting An Input Icon In A Palette Representing Input Utility.”

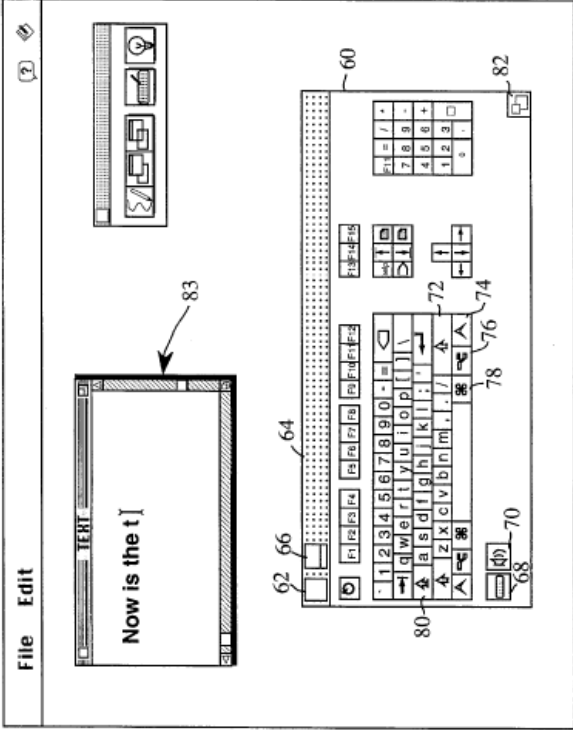
⁴ Upon information and belief, the MessagePad 120 was first sold in the United States in January of 1995. The Newton Apple MessagePad Handbook would have been distributed to the public around the same time.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt ² and/or U.S. Patent No. 5,603,053 ³ and/or the MessagePad 120 ⁴
	list that includes one or more selectable input methods for one or more computer programs, wherein each input method is a computer-executable software component distinct from the computer programs;	<p>present invention, e.g. handwriting “recognition” windows, etc.”).</p> <p>These overlays are executable independent of the application programs. Col. 2, ll. 29-33 (“Preferably, the base image is produced by an unmodified application program running on the computer system, and the overlay image is produced by a computer implemented process of the present invention referred to herein as the “overlay utility”.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55.</p> <p>The Gough ‘053 describes a method for “displaying a palette having at least one icon representing an input utility, and selecting the icon to create an input image.” Gough ‘053 Col. 3, ll. 15-17. In one embodiment of the Gough ‘053 patent, the palette can be set to launch itself “upon the start-up of the computer system[.]” Gough ‘053 Col. 6, ll. 12-13.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to display an actuable icon representative of an input method list to the user. For example, Gough ‘501 displays palette 46, and it would have been obvious to one of skill in the art to display palette 46 in response to a user request. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the display of an actuable icon representative of an input method list to the user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified,</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt ² and/or U.S. Patent No. 5,603,053 ³ and/or the MessagePad 120 ⁴
		predictable solutions.
1.2	in response to actuation of the actuatable icon, displaying the input method list;	<p>See Claim 1.1 regarding the actuation of the actuatable icon.</p> <p>Regarding the display of the input method list, these input methods are shown by the utility palette.</p> <p>See Figure 3.</p>  <p>The Gough ‘053 describes a method for “displaying a palette having at least one icon representing an input utility, and selecting the icon to create an input image.” Gough ‘053 Col. 3, ll. 15-17. In one embodiment of the Gough ‘053 patent, the palette can be set to launch itself “upon the start-up of the computer system[.]” Gough ‘053 Col. 6, ll. 12-13.</p>
1.3	receiving a selection of an input method from the input method list;	<p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>Figures 4a and 4b of the Gough ‘053 patent demonstrate the selection (by dragging) of the keyboard image from the input method list.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt ² and/or U.S. Patent No. 5,603,053 ³ and/or the MessagePad 120 ⁴
		 <p style="text-align: center;">Figure 4a</p>  <p style="text-align: center;">Figure 4c</p>
1.4	installing an input method component that corresponds to the selected input method, the input method component causing an interactive input panel to be displayed;	<p>“Preferably, this keyboard image 64 is provided by dragging a keyboard icon 66 off of the PenBoard palette 46 in a fashion more fully described in copending U.S. patent application Ser. No. 08/060,458, filed May 10, 1993, on behalf of Gough et al., entitled “Method and Apparatus for Interfacing With a Computer System”, and assigned to the assignee of the present application, the disclosure of which is hereby incorporated herein by reference in its entirety.” Col. 6, ll. 18-26.</p> <p>Once installed, the input method may communicate with the active application. See</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt² and/or U.S. Patent No. 5,603,053³ and/or the MessagePad 120⁴</p>
		<p>Claim 1 of Gough ‘501 (“[I]n response to said translucency icon being selected, translucently displaying said overlay image[.]”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Claim 1 of Gough ‘053 (“A method for launching an input utility . . . said method comprising . . . activating said input utility and displaying an input image created for said input utility at about said desired location, said input image dedicating a portion of said screen for data input into said input utility; and linking said input image to said active application such that data input into said input utility is communicated to said active application.”).</p>
1.5	receiving input via the interactive input panel; and	<p>“Information can be entered into the window 44 of the application program from the keyboard image 64 by ‘tapping’ on a ‘key’ with the stylus 38. For example, arrow 68 on the keyboard image 64 represents the “tapping” on the key “R” with the stylus 38. This tapping action will send a “R” to be displayed in the window 44 of the AppleShare application just as if a “R” had been typed on a physical keyboard.” Col. 6, ll. 30-37.</p> <p>See Claim 3 of Gough ‘501 (“A method as recited in claim 2 wherein said second computer implemented process intercepts screen inputs which contact said overlay image and processes said screen inputs.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Claim 3 of Gough ‘053 (“A method for inputting data to an active application . . . the method comprising . . . detecting the engagement of said input image by a pointer means; analyzing said engagement to determine input data; and sending said input data to said active application program.”).</p> <p>See Figure 4c of Gough ‘053, which shows active application receiving input data</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt² and/or U.S. Patent No. 5,603,053³ and/or the MessagePad 120⁴</p> <p>from the input panel.</p>
		 <p style="text-align: center;"><i>Figure 4c</i></p>
1.6	<p>providing the input to a computer program of the one or more computer programs as if the information was received via user input received from a hardware input device.</p>	<p>“This tapping action will send a “R” to be displayed in the window 44 of the AppleShare application just as if a “R” had been typed on a physical keyboard.” Col. 6, ll. 34-37.</p> <p>See Claim 6 of Gough ‘501 (“A method as recited in claim 3 wherein said second computer implemented process is further operative to update said first computer implemented process according to the step of processing said screen inputs.”).</p> <p>The prior art includes input windows that accept user input and transmit that input to programs “as if it were typed from a keyboard.” Col. 2, ll. 5-8. The improvement of this invention is that the input window is transparent. Col. 2, ll. 15-17 (“The overlay image can serve as an input device for application programs without obscuring images</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt² and/or U.S. Patent No. 5,603,053³ and/or the MessagePad 120⁴</p>
		<p>made on the screen by the application programs.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Claim 3 of Gough ‘053 (“A method for inputting data to an active application . . . the method comprising . . . detecting the engagement of said input image by a pointer means; analyzing said engagement to determine input data; and sending said input data to said active application program.”).</p> <p>See Figure 4c of Gough ‘053, which shows active application receiving input data from the input panel.</p>

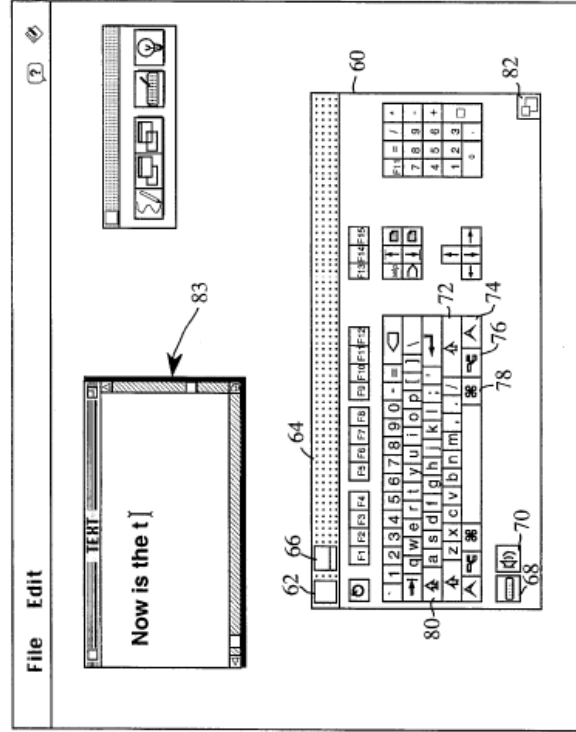


Figure 4c

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt ² and/or U.S. Patent No. 5,603,053 ³ and/or the MessagePad 120 ⁴

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
2	2. The method of claim 1 wherein providing the input to the computer program comprises communicating information representative of the input to a graphical windowing environment.	See Figures 3-5. “In FIG. 2, a screen 40 of a Macintosh computer system made by Apple Computer, Inc., of Cupertino, Calif., includes a desktop image 42 produced by a Macintosh operating system[.]” Col. 5, ll. 48-51. Compare Col. 8, ll. 26-30 (“Running under the operating system 96 is an application program 98, such as the aforementioned AppleShare application program. Application program 98, when it wants to open a window such as window 44 of FIG. 2, calls a set of routines 100 provided by the operating system 96.”) with Col. 7, ll. 1-5 (“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program and be displayed in a password field of the window 44.”).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
3	3. The method of claim 2 wherein communicating the information comprises passing the information to an interface.	“It will be apparent with a study of FIGS. 4 and 5a-5c that the translucent keyboard image 64’ is a distinctly superior user <i>interface</i> for situations in which screen area is at a premium. Since images “beneath” the translucent keyboard image 64’ can be seen through the keyboard image, the user has immediate feedback as to the accuracy of his

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120</p>
		<p>or her input to the active application program.” Col. 8, ll. 23-26 (emphasis added).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Figure 5 of Gough ‘053.</p> <div data-bbox="602 333 953 1094" data-label="Diagram"> <p style="text-align: center;">Figure 5</p> <pre> graph TD 85((APPLICATION)) <--> 84[PENBOARD APPLICATION] 85 --> 87[EVENT QUEUE] 87 --> 84 </pre> </div> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system capable of calling functions in the manager component via a defined interface set. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and U.S. Patent 5,157,384 (“Greanias ‘384”) renders this claim obvious. Greanias ‘384 teaches calling functions in the manager component via a defined interface set. See Greanias ‘384 Claim 37 and Col. 8, ll. 59-65. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120	combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120	
4	4. The method of claim 2 further comprising, communicating the information from the graphical windowing environment to an application program, wherein the computer program includes the application program, wherein the information is provided to the application program in a same manner as if the input was received via a hardware keyboard.	<p>“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program and be displayed in a password field of the window 44.” Col. 7, ll. 1-5.</p> <p>The prior art includes input windows that accept user input and transmit that input to programs “as if it were typed from a keyboard.” Col. 2, ll. 5-8.</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55.</p> <p>The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Col. 7, ll. 39-44 of Gough ‘053 (“[T]he computer implemented process of the present invention intercepts calls made by TeachText to the input event queue and passes characters to TeachText from the keyboard image 60 just as if they had been entered into the computer system from a physical keyboard.”).</p>	

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
6	6. The method of claim 1 wherein the selected input method corresponds to a displayed keyboard, and wherein receiving input via the interactive input panel that corresponds to the selected input method comprises receiving information corresponding to a keyboard character entered via the displayed keyboard.	“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program and be displayed in a password field of the window 44.” Col. 7, ll. 1-5.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
8	8. The method of claim 1 further comprising, hiding the input panel.	<p>See Figures 3 & 6b.</p> <p>“The process 80 begins at 84, and in a step 86, it is determined whether the process 80 is completed. In this instance, <i>a process 80 is completed when the “button” 69 of the translucent keyboard image 64’ (see FIG. 4) is tapped</i>. If the process is completed, the overlay utility 80 is <i>terminated</i> as indicated at 88. If the process is not completed, a step 90 displays an “overlay” image on the screen such that images on the screen that it overlaps can be seen through the overlay image.” Col. 7, ll. 50-58 (emphasis added).</p> <p>See Claim 20 of Gough ‘501 (“A pen computer system comprising . . . means coupled to said screen means for displaying an overlay image . . . said means for displaying said overlay image responsive to a users selection to toggle between (i) opaquely displaying said overlay image and (ii) translucently displaying said overlay image.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”). The Gough ‘053 patent permits utility input windows to be “hidden from view when not required.” Col. 3, ll. 48 of Gough ‘053.
Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
9	9. The method of claim 1 further comprising, docking the input panel.	<p>“[T]he palette 46 (<i>which is just a specialized form of window</i>) is produced by the PenBoard application, and does not occupy the entire space of the screen 40.” Col. 6, ll. 4-7 (emphasis added).</p> <p>See Claim 8 of Gough ‘501 (“A method as recited in claim 1 wherein said computer system is a pointer based computer system, said first utility icon is selected by dragging said first utility icon off of said utility palette with said pointer, and said overlay image is displayed in a location correlated to where said first utility icon is dragged.”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need</p>

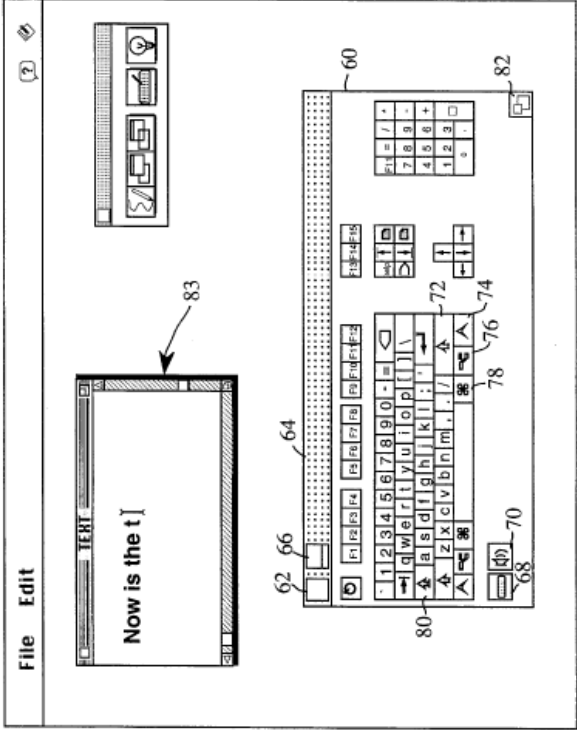
Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

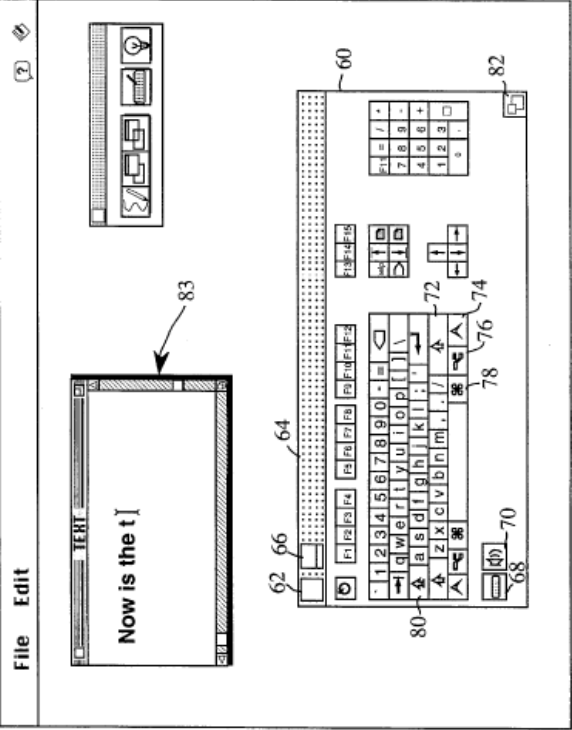
Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
10	10. At least one computer-readable medium having computer-executable instructions, which when executed perform the method of claim 1.	<p>“As shown in FIG. 1, a pen computer system 10 in accordance with the present invention includes a central processing unit (CPU) 12, read only memory (ROM) 14, random access memory (RAM) 16, expansion RAM 17, input/output (I/O) circuitry 18, display assembly 20, and expansion bus 22. The pen computer system 10 may also optionally include a mass storage unit 24 such as a disk drive unit or nonvolatile memory such as flash memory and a real-time clock 26.” Col. 4, ll. 32-40.</p> <p>“A pen computer system in accordance with the present invention includes a central processing unit (CPU), a screen assembly coupled to the CPU, a mechanism coupled to the screen assembly for displaying a base image on the screen assembly, and a mechanism coupled to the screen assembly for displaying an overlay image on the screen assembly such that portions of the base image which are overlapped by the overlay image are at least partially visible through the overlay image.” Col. 2, ll. 54-62.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
11	11. At least one computer-readable medium having computer-executable instructions stored thereon, which when executed by a computer system perform steps, comprising:	<p>“As shown in FIG. 1, a pen computer system 10 in accordance with the present invention includes a central processing unit (CPU) 12, read only memory (ROM) 14, random access memory (RAM) 16, expansion RAM 17, input/output (I/O) circuitry 18, display assembly 20, and expansion bus 22. The pen computer system 10 may also optionally include a mass storage unit 24 such as a disk drive unit or nonvolatile memory such as flash memory and a real-time clock 26.” Col. 4, ll. 32-40.</p> <p>“A pen computer system in accordance with the present invention includes a central processing unit (CPU), a screen assembly coupled to the CPU, a mechanism coupled to the screen assembly for displaying a base image on the screen assembly, and a mechanism coupled to the screen assembly for displaying an overlay image on the screen assembly such that portions of the base image which are overlapped by the overlay image are at least partially visible through the overlay image.” Col. 2, ll. 54-62.</p>
11.1	selecting one of a plurality of executable input methods for supplying user input to the computer system, wherein each executable input method is an interchangeable software component distinct from one or more application programs, each executable input method having a defined interface set such that the executable input method is connectable to the application programs;	<p>The invention provides for multiple types of overlays to be used as inputs. Col. 7, ll. 60-63 (“Of course, other overlay images besides keyboards can be provided by the present invention, e.g. handwriting “recognition” windows, etc.”).</p> <p>These overlays can be used with multiple programs. Col. 3, ll. 23-28 (“Another advantage of the overlay image of the present invention is that it works with both pen-aware and non-pen-aware application programs. Therefore, the overlay image of the present invention can be used with the many thousands of application programs which are not designed to be used in pen computer systems.”).</p> <p>These overlays are executable independent of the application programs. Col. 2, ll. 29-33 (“Preferably, the base image is produced by an unmodified application program running on the computer system, and the overlay image is produced by a computer implemented process of the present invention referred to herein as the “overlay utility”.”).</p> <p>The overlays are connectable to the application programs. Col. 2, ll. 42-47 (“Preferably, the step of running the overlay program includes the steps of. . . 2) intercepting screen inputs which contact the overlay image; 3) processing the</p>

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120</p>
		<p>intercepted screen inputs in the CPU; and 4) updating the application program based upon the process screen inputs.’”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>Figures 4a and 4b of the Gough ‘053 patent demonstrate the selection (by dragging) of the keyboard image from the input method list.</p> <div data-bbox="634 520 963 934"> </div> <p style="text-align: center;"><i>Figure 4a</i></p> <div data-bbox="1019 495 1386 963"> </div> <p style="text-align: center;"><i>Figure 4c</i></p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system capable of calling functions in the manager component via a defined interface set. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and U.S. Patent 5,157,384 (“Greanias ‘384”) renders this claim obvious. Greanias ‘384 teaches calling functions in the manager component via a defined interface set. See Greanias ‘384 Claim 37 and Col. 8, ll. 59-65. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
11.2	opening an input window on a display of the computer system independent of a window of an active application program; and	<p>The overlay utility opens an input window. Col. 2, ll. 42-44 (“Preferably, the step of running the overlay program includes the step[] of . . . displaying an overlay image on the screen[.]”).</p> <p>These overlays are executable independent of the application programs. Col. 2, ll. 29-33 (“Preferably, the base image is produced by an unmodified application program running on the computer system, and the overlay image is produced by a computer implemented process of the present invention referred to herein as the “overlay utility”.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Figure 4c of Gough ‘053, which shows independent windows for the palette and keyboard input window.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 ("Gough '501") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		 <p style="text-align: center;"><i>Figure 4c</i></p>
11.3	displaying an interactive input panel in the input window, the interactive input panel corresponding to the selected executable input method such that information corresponding to user input received by the selected executable input method via the interactive input panel is provided to the active application program as if the information was received via user input at a hardware input device.	<p>See Figures 3-5.</p> <p>See Col. 6, ll. 26-37 discussing the prior art opaque overlays ("As can be seen in this FIG. 3, the keyboard image 64 completely obscures the icons 52, 54 and 56 of FIG. 2, and almost totally obscures the window 44 of the AppleShare application program. Information can be entered into the window 44 of the application program from the keyboard image 64 by "tapping" on a "key" with the stylus 38. For example, arrow 68 on the keyboard image 64 represents the "tapping" on the key "R" with the stylus 38. This tapping action will send a "R" to be displayed in the window 44 of the AppleShare application just as if a "R" had been typed on a physical keyboard.").</p> <p>The Gough '501 patent utilizes a "palette 46 produced by a small application program or 'utility' known as 'PenBoard' made by Apple Computer, Inc." Col. 5, ll. 53-55.</p>

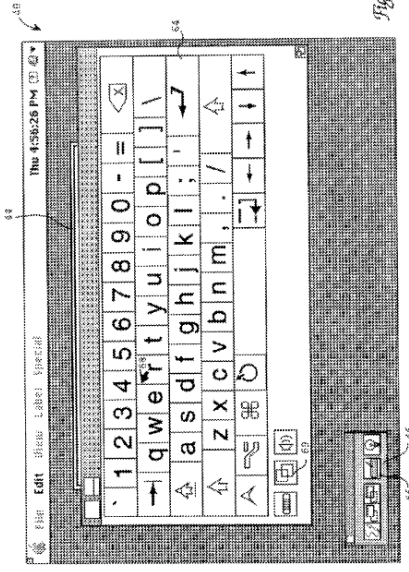
Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120</p>
		<p>The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Claim 3 of Gough ‘053 (“A method for inputting data to an active application . . . the method comprising . . . detecting the engagement of said input image by a pointer means; analyzing said engagement to determine input data; and sending said input data to said active application program.”).</p> <p>See Figure 4c of Gough ‘053, which shows: (1) independent windows for the palette and keyboard input window; and (2) an active application receiving input data from the input panel.</p>
		 <p style="text-align: center;"><i>Figure 4c</i></p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
13	13. The computer-readable medium of claim 11 further comprising, providing a Software Input Panel (SIP) menu button on the display of the computer system, the SIP menu button being actuatable to display a selectable list of the plurality of executable input methods.	<p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The Gough ‘053 describes a method for “displaying a palette having at least one icon representing an input utility, and selecting the icon to create an input image.” Gough ‘053 Col. 3, ll. 15-17. In one embodiment of the Gough ‘053 patent, the palette can be set to launch itself “upon the start-up of the computer system[.]” Gough ‘053 Col. 6, ll. 12-13.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to display an actuatable icon representative of an input method list to the user. For example, Gough ‘501 displays palette 46, and it would have been obvious to one of skill in the art to display palette 46 in response to a user request. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough ‘501 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the display of an actuatable icon representative of an input method list to the user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 ("Gough '501") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
14	14. The computer-readable medium of claim 13 further comprising, receiving a selection of one of the plurality of executable input methods displayed in the list as a selected executable input method, and in response, closing any open input window, and opening a new input window corresponding to the selected executable input method.	<p>The overlay utility opens an input window. Col. 2, ll. 42-44 ("Preferably, the step of running the overlay program includes the step[] of . . . displaying an overlay image on the screen[.]").</p> <p>The Gough '501 patent utilizes a "palette 46 produced by a small application program or 'utility' known as 'PenBoard' made by Apple Computer, Inc." Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 ("Gough '053").</p> <p>See Figure 4c of Gough '053, which shows independent windows for the palette and keyboard input window.</p> <div data-bbox="678 365 1250 1092"> </div> <p><i>Figure 4c</i></p> <p>The PenBoard utility is described by U.S. Patent No. 5,603,053 ("Gough '053"). The Gough '053 patent permits utility input windows to be "hidden from view when not required." Col. 3, ll. 48 of Gough '053.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to hide the input panel upon user command or upon the opening of another input window. Col. 14, ll. 23-29.
		In addition, the combination of Gough ‘501 and U.S. Patent No. 5,148,155 (Martin ‘155) renders this claim obvious. Martin ‘155 teaches the hiding of an input panel. See Martin ‘155 Abstract and Col. 17, ll. 11-22. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem, disclose the same or similar techniques, and were developed during the same general time period.
		Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
15	15. At least one computer-readable medium having computer-executable instructions, which when executed perform steps, comprising:	“As shown in FIG. 1, a pen computer system 10 in accordance with the present invention includes a central processing unit (CPU) 12, read only memory (ROM) 14, random access memory (RAM) 16, expansion RAM 17, input/output (I/O) circuitry 18, display assembly 20, and expansion bus 22. The pen computer system 10 may also optionally include a mass storage unit 24 such as a disk drive unit or nonvolatile memory such as flash memory and a real-time clock 26.” Col. 4, ll. 32-40.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 ("Gough '501") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
15.1	presenting icons corresponding to a plurality of input methods available for a computer application, wherein each input method is a computer-executable software component distinct from the computer application;	<p>The invention provides for multiple types of overlays to be used as inputs. Col. 7, ll. 60-63 ("Of course, other overlay images besides keyboards can be provided by the present invention, e.g. handwriting "recognition" windows, etc.").</p> <p>These overlays are executable independent of the application programs. Col. 2, ll. 29-33 ("Preferably, the base image is produced by an unmodified application program running on the computer system, and the overlay image is produced by a computer implemented process of the present invention referred to herein as the "overlay utility".).</p> <p>The Gough '501 patent utilizes a "palette 46 produced by a small application program or 'utility' known as 'PenBoard' made by Apple Computer, Inc." Col. 5, ll. 53-55.</p> <p>Figure 3 of the Gough '501 patent shows the utility palette of the invention which presents icons representative of the various input methods.</p>  <p>To the extent not expressly or inherently disclosed, it would have been obvious to display the input method list in response to actuation of the actuatable icon. For example, Gough '501 displays palette 46, and it would have been obvious to one of skill in the art to display palette 46 in response to a user request. Col. 9, ll. 64-67 to Col. 10, ll. 1-3 of Gough '053.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
15.2	invoking a selected input method in response to a user selecting an icon corresponding to the selected input method, including presenting an input panel window; and	<p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>Figures 4a and 4b of the Gough ‘053 patent demonstrate the selection (by dragging) of the keyboard image from the input method list, and the presentation of the keyboard input panel window.</p> <div data-bbox="576 520 906 934"> </div> <p style="text-align: center;"><i>Figure 4a</i></p> <div data-bbox="961 495 1331 963"> </div> <p style="text-align: center;"><i>Figure 4c</i></p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
15.3	accepting user data entered in the input panel window for the computer application, wherein the user data is provided to the computer application as if the user data was received from a hardware input device.	<p>“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program and be displayed in a password field of the window 44.” Col. 7, ll. 1-5.</p> <p>See Claim 3 of Gough ‘501 (“A method as recited in claim 2 wherein said second computer implemented process intercepts screen inputs which contact said overlay image and processes said screen inputs.”).</p> <p>See Claim 6 of Gough ‘501 (“A method as recited in claim 3 wherein said second computer implemented process is further operative to update said first computer implemented process according to the step of processing said screen inputs.”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Claim 3 of Gough ‘053 (“A method for inputting data to an active application . . . the method comprising . . . detecting the engagement of said input image by a pointer means; analyzing said engagement to determine input data; and sending said input data to said active application program.”).</p> <p>See Figure 4c of Gough ‘053, which shows active application receiving input data from the input panel as if typed from a normal keyboard.</p>

Claim	U.S. Patent 5,638,501 ("Gough '501") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120	U.S. Patent 7,411,582
	<p style="text-align: center;">Figure 4c</p>	

Claim	U.S. Patent 5,638,501 ("Gough '501") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120	U.S. Patent 7,411,582
16	<p>U.S. Patent 5,638,501 ("Gough '501") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120</p> <p>"A method and apparatus for providing a translucent overlay image over a base image on the screen of a computer system." Abstract.</p> <p>Like the prior art, the transparent overlay keyboard "provide[s] a recognition window for inputting handwritten data which is then recognized and sent to an application</p>	16. The computer-readable medium of claim 15 wherein accepting user data includes detecting user interaction with a touch-sensitive display.

Claim	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
	program as if it were typed from a keyboard[.]” Col. 2, ll. 6-8.

Claim	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
17	<p>17. The computer-readable medium of claim 15 wherein each input method comprises a component object model (COM) object, and wherein the step of invoking the selected input method includes the step of instantiating the COM object.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 14, ll. 23-29. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Gough ‘501 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
18	18. The computer-readable medium of claim 15 further comprising converting the user data to a Unicode character value.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convert the user data to a Unicode character value. Col. 14, ll. 23-29. In addition, the combination of Gough ‘501 and U.S. Patent No. 5,454,046 (“the Carmen Reference”) ⁵ renders this claim obvious. See Carmen Reference, Col. 10, ll. 14-22 (“The text represented at block 62 may consist of one or more words represented in Unicode, a 16-bit character representation defined by the Unicode Consortium of Mountain View, Calif. This representation is used instead of ASCII or ANSI representations, since Unicode can accommodate over 65,000 character representations, representing the definition of character glyphs of virtually all

⁵ “Universal Symbolic Handwriting Recognition System” issued to Carman; filed September 17, 1993; issued September 26, 1995.

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120</p>
		<p>languages of the world, whereas ASCII and ANSI representations are limited to 256 characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>See Figure 3A of Carmen Reference.</p> <div data-bbox="662 478 1094 989"> </div> <p>FIG. 3A</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified,</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		<p>predictable solutions.</p> <p>Similarly, the combination of Gough ‘501 and U.S. Patent No. 5,455,901 (“the Friend Reference”)⁶ renders this claim obvious. See Claim 6 of Friend Reference (“The device of claim 5 wherein said translator comprises a generator for generating machine readable characters from said strokes.”); Claim 7 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise Unicode characters.”); Claim 8 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise ASCII characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
19	19. In a computing environment, a	“This invention relates generally to computer systems, and more particularly to

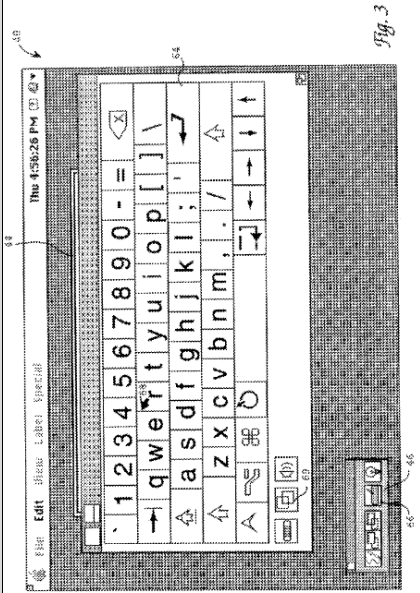
⁶ “Input Device With Deferred Translation” issued to Friend; filed September 12, 1994; issued October 3, 1995.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
	system comprising,	graphical user interfaces for computer systems.” Col. 1, ll. 5-7.
19.1	a manager component stored on one or more computer-readable media and configured: to manage selection of a selected input method from one or more available stored input methods, wherein each input method is a computer-executable software component distinct from one or more computer programs, and to send input data corresponding to a user input received at the selected input method to a graphical windowing environment; and	<p>The input method utility program serves as the manager component to interface between the user input and the application [here, Appleshare] and send the user input to the application window. Col 7, ll. 1-5 (“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program [the open application] and be displayed in a password field of the window 44.”).</p> <p>See Figure 6b. See also Col. 7, ll. 63-65 (“Next, in a step 92, the overlay utility intercepts screen inputs which contact the overlay image, and these screen inputs are processed.”).</p> <p>The invention provides for multiple types of overlays to be used as inputs. Col. 7, ll. 60-63. These overlays can be used with multiple programs. Col. 3, ll. 23-28. These overlays are executable independent of the application programs. Col. 2, ll. 29-33.</p>
19.2	the graphical windowing environment to receive the input data and to send the input data to a computer program of the one or more computer programs, wherein the input data is sent to the computer program as if the input data was received via user input received from a hardware input device.	<p>The application receives the input data – in its application window – from the utility application as if were entered on a keyboard. See Claim 15 of Gough ‘501 (A method as recited in claim 14 wherein said step of running an overlay program comprises the steps of: intercepting screen inputs which contact said translucent overlay image; processing said intercepted screen inputs in said CPU; and updating said application program based upon said processed screen inputs.”); see also Col. 2, ll. 6-8 (Like the prior art, the transparent overlay keyboard “provide[s] a recognition window for inputting handwritten data which is then recognized and sent to an application program as if it were typed from a keyboard[.]”).</p>

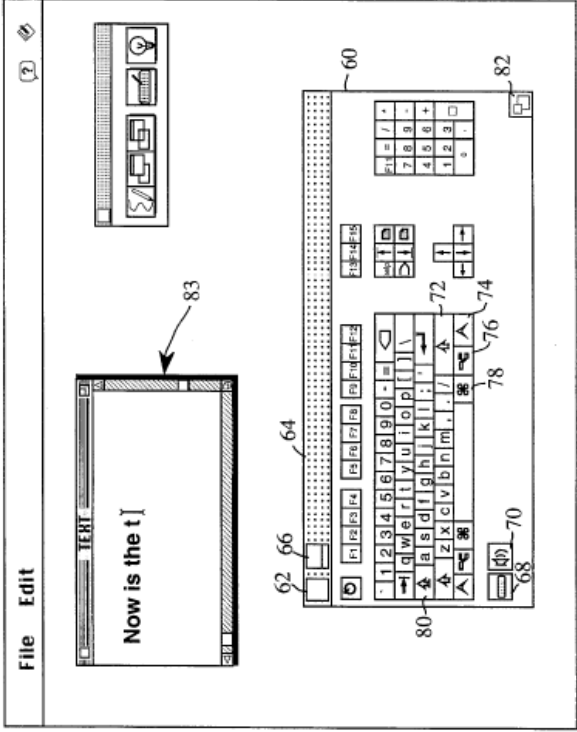
Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
20	20. The system of claim 19 wherein the computer program comprises an application program having focus.	“[T]he keyboard image 64 can be used to input data into a currently active application program (such as AppleShare)[.]” Col. 6, ll. 40-42.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
21	21. The system of claim 19 further comprising an input panel window corresponding to the selected input method.	See Figures 3-5. “The present invention provides a transparent overlay image over a base image provided on a screen of a pen computer system. The overlay image can serve as an input device for application programs without obscuring images made on the screen by the application programs.” Col. 6, ll. 12-16.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
22	22. The system of claim 21 wherein the selected input method presents an image representing a keyboard on the input panel window.	See Claim 7 of Gough ‘501 (“A method as recited in claim 3 wherein said overlay image is a keyboard image comprising icons which represent alphanumeric characters”). See Figure 3.

Claim	U.S. Patent 5,638,501 (“Gough ‘501’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120	
	 <p style="text-align: right;">Fig. 3</p>	

Claim	U.S. Patent 5,638,501 (“Gough ‘501’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120	
23	<p>The overlay utility opens an input window. Col. 2, ll. 42-44 (“Preferably, the step of running the overlay program includes the step[] of . . . displaying an overlay image on the screen[.]”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”). See Figure 4c of Gough ‘053, which shows independent windows for the palette and keyboard input window.</p>	

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 ("Gough '501") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		 <p style="text-align: center;"><i>Figure 4c</i></p> <p>The PenBoard utility is described by U.S. Patent No. 5,603,053 ("Gough '053"). The Gough '053 patent permits utility input windows to be "hidden from view when not required." Col. 3, ll. 48 of Gough '053.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to hide the input panel upon user command or upon the opening of another input window. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough '501 and U.S. Patent No. 5,148,155 (Martin '155) renders this claim obvious. Martin '155 teaches the hiding of an input panel. See Martin '155 Abstract and Col. 17, ll. 11-22. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		program), address the same problem, disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
25	25. The system of claim 19 where the input method is displayed on a touch-sensitive display screen.	“A method and apparatus for providing a translucent overlay image over a base image on the screen of a computer system.” Abstract. Like the prior art, the transparent overlay keyboard “provide[s] a recognition window for inputting handwritten data which is then recognized and sent to an application program as if it were typed from a keyboard[.]”. Col. 2, ll. 6-8.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
26	26. The system of claim 19 wherein the manager component transfers	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design an input manager capable of transferring information from

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
	information from the computer program to the selected input method.	the computer program to the selected input method. Col. 14, ll. 23-29.
		In addition, the combination of Gough ‘501 and U.S. Patent No. 5,148,155 (Martin ‘155) renders this claim obvious. Martin ‘155 teaches the transferring information from the computer program to the selected input method. See Martin ‘155 Col. 32, ll. 36-44. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
27	27. The system of claim 19 wherein the selected input method calls functions in the manager component via a defined interface set.	The input method utility program serves as the manager component to interface between the user input and the application [here, Appleshare] and send the user input to the application window. Col 7, ll. 1-5 (“The keyboard image 64’ “intercepts” the tap 68 which would otherwise fall on the window 44, and, instead causes a “r” to be sent to the AppleShare program [the open application] and be displayed in a password field of the window 44.”). To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system capable of calling functions in the manager

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		component via a defined interface set. Col. 14, ll. 23-29. In addition, the combination of Gough ‘501 and U.S. Patent 5,157,384 (“Greanias ‘384”) renders this claim obvious. Greanias ‘384 teaches calling functions in the manager component via a defined interface set. See Greanias ‘384 Claim 37 and Col. 8, ll. 59-65. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

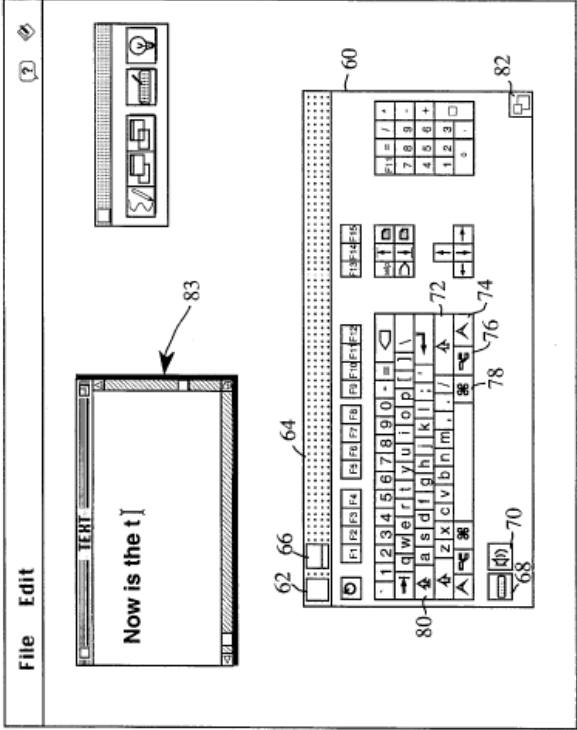
Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
28	28. The system of claim 19 wherein the selected input method comprises an object.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 14, ll. 23-29. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).
		In addition, the combination of Gough ‘501 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.
		Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
29	29. The system of claim 19 wherein the selected input method draws an input	See Figures 3-5.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
	panel in an input panel window displayed in the graphical windowing environment.	

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
30	30. The system of claim 29 wherein the manager component selectively displays and hides the display of the input panel window.	<p>The overlay utility opens an input window. Col. 2, ll. 42-44 (“Preferably, the step of running the overlay program includes the step[] of . . . displaying an overlay image on the screen[.]”).</p> <p>The Gough ‘501 patent utilizes a “palette 46 produced by a small application program or ‘utility’ known as ‘PenBoard’ made by Apple Computer, Inc.” Col. 5, ll. 53-55. The PenBoard utility is described by U.S. Patent No. 5,603,053 (“Gough ‘053”).</p> <p>See Figure 4c of Gough ‘053, which shows independent windows for the palette and keyboard input window.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 ("Gough '501") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		 <p style="text-align: center;"><i>Figure 4c</i></p> <p>The PenBoard utility is described by U.S. Patent No. 5,603,053 ("Gough '053"). The Gough '053 patent permits utility input windows to be "hidden from view when not required." Col. 3, ll. 48 of Gough '053.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to hide the input panel upon user command or upon the opening of another input window. Col. 14, ll. 23-29.</p> <p>In addition, the combination of Gough '501 and U.S. Patent No. 5,148,155 (Martin '155) renders this claim obvious. Martin '155 teaches the hiding of an input panel. See Martin '155 Abstract and Col. 17, ll. 11-22. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		program), address the same problem, disclose the same or similar techniques, and were developed during the same general time period.
		Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
31	31. The system of claim 29 wherein the manager component docks the input panel window.	<p>“[T]he palette 46 (<i>which is just a specialized form of window</i>) is produced by the PenBoard application, and does not occupy the entire space of the screen 40.” Col. 6, ll. 4-7 (emphasis added).</p> <p>See Claim 8 of Gough ‘501 (“A method as recited in claim 1 wherein said computer system is a pointer based computer system, said first utility icon is selected by dragging said first utility icon off of said utility palette with said pointer, and said overlay image is displayed in a location correlated to where said first utility icon is dragged.”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 9, ll. 64-67 to Col. 10, ll. 1-3 of Gough ‘053.</p> <p>In addition, the combination of Gough ‘501 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,638,501 (“Gough ‘501”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Brockschmidt and/or U.S. Patent No. 5,603,053 and/or the MessagePad 120
		<p>and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

EXHIBIT H

Exemplar Chart of U.S. Patent 7,411,582

**U.S. Patent 5,157,384 (“Greanias ‘384”)¹
Claims 1-4, 6, 8-11, 13-23, 25-31**

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
1	1. In a computing environment, a computer-implemented method comprising:	“An advanced user interface for use with a computer system operating on an integrated operating environment.” Abstract.
1.1	displaying an actuatable icon representative of an input method list that includes one or more selectable input methods for one or more computer programs, wherein each input method is a computer-executable software component distinct from the computer programs;	<p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract.</p> <p>“It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 30-32.</p> <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch</p>

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Advanced User Interface” issued to Greanias et al.; filed April 28, 1989; issued October 20, 1992.

² Kraig Brockschmidt, Inside OLE (2d ed. 1995).

³ Upon information and belief, the MessagePad 120 was first sold in the United States in January of 1995. The Newton Apple MessagePad Handbook would have been distributed to the public around the same time.

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt² and/or the MessagePad 120³</p> <p>input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i>, the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 14-23 (emphasis added).</p> <p>The input methods are each separately executable from the application programs. Col. 7, ll. 25 (“the AUI is written as an application program”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement an actuable icon representative of an input method list. For example, the invention discloses several types of interface utilities 109, and it would have been obvious to one of skill in the art to include this step to permit the user to select one of those input utilities. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p> <p>In addition, the combination of Greanias ‘384 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the presentation of an actuable icon representative of an input method list to a user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.2	in response to actuation of the actuable icon, displaying the input method list;	<p>See Claim 1.1 regarding the actuation of the actuable icon.</p> <p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract.</p> <p>“It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 30-32.</p> <p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon,</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
		the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to present an input list to a user. Col. 3, ll. 13-27; Col. 8, ll. 13-27. In addition, the combination of Greanias ‘384 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the presentation of an input list to a user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.
1.3	receiving a selection of an input method from the input method list;	See Claim 1.2.
1.4	installing an input method component that corresponds to the selected input method, the input method component causing an interactive input panel to be displayed;	A user can install an input method by using the appropriate method. Col. 8, ll. 20-23. (“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.”). Once installed, the input method can use the translation capabilities of the AUI 100 to communicate with the active application. Col. 7, ll. 15. See also Col. 8, ll. 23-30 (“A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard. The space needed for the keyboard or keypad is quite small, and will not completely obscure the work area of the spreadsheet, word processor, etc., which is the active application program.”).
1.5	receiving input via the interactive input panel; and	Once installed, the input method can use the translation capabilities of the AUI 100 to communicate with the active application. Col. 7, ll. 15. See also Col. 8, ll. 23-30 (“A

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
		user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard. The space needed for the keyboard or keypad is quite small, and will not completely obscure the work area of the spreadsheet, word processor, etc., which is the active application program.”).
1.6	providing the input to a computer program of the one or more computer programs as if the information was received via user input received from a hardware input device.	<p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code. The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface.” Abstract.</p> <p>“The alternative input subsystem module provides communication between the attached user friendly input devices and the remainder of the advanced user interface as well as application programs through the integrated operating environment. The alternative input subsystem module receives the input signals generated by the input devices and translates them to input messages useable in the advanced user interface.” Col. 3, ll. 57-65.</p> <p>The AUI manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs. Col. 7, ll. 15.</p> <p>“After receiving an input message from Presentation Manager.TM. 156 together with information about which is the active application program, the PM-Link 201 refers back to the alternate input subsystem 203 to determine whether the “keyboard” or “mouse” message it received is in fact a keyboard or a mouse message, or rather a touch, voice, image, or other message. The PM-Link 201 then passes the true message to the appropriate application program.” Col. 13, ll. 16-25.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
2	2. The method of claim 1 wherein providing the input to the computer program comprises communicating information representative of the input to a graphical windowing environment.	“As shown [in Figure 3], the AUI utility 109 can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.” Col. 8, ll. 8-13.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
3	3. The method of claim 2 wherein communicating the information comprises passing the information to an interface.	<p>“The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.” Abstract.</p> <p>“The alternative input subsystem module provides communication between the attached user friendly input devices and the remainder of the advanced user interface as well as application programs through the integrated operating environment. The alternative input subsystem module receives the input signals generated by the input devices and translates them to input messages useable in the advanced user interface.” Col. 3, ll. 57-65.</p> <p>The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs.” Col. 7, ll. 15.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
4	4. The method of claim 2 further comprising, communicating the information from the graphical windowing environment to an application program, wherein the computer program includes the application program, wherein the information is provided to the application program in a same manner as if the input was received via a hardware keyboard.	<p>The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs.” Col. 7, ll. 15.</p> <p>“Another recent trend is to provide some sort of integration of computer program applications. Without integration, the user must employ separate application programs for word processing, database manipulation, graphics and electronic mail functions, and so forth. It is often quite difficult to integrate the outputs of the different programs into a single desired output. One solution has been to write a single integrated piece of software which incorporates a variety of applications which is called a multiple-function program.” Col. 2, ll. 35-44.</p> <p>“The interface profile module 104 is comprised of sets of application profiles 105 and the user profiles 107, which are files which list input messages produced by the AIS 103 from the input signals received by input devices 36, 38, 40, mapped to keyboard, mouse or other commands which are usable by existing application programs, e.g., mouse clicks, keystroke messages, MACROs, utility programs, etc.” Col. 7, ll. 66-68 to Col. 8, ll. 1-5.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
6	6. The method of claim 1 wherein the selected input method corresponds to a displayed keyboard, and wherein receiving input via the interactive input panel that corresponds to the selected input method comprises receiving information corresponding to a keyboard	<p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.” Col. 8, ll. 20-27.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
	character entered via the displayed keyboard.	

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
8	8. The method of claim 1 further comprising, hiding the input panel.	<p>See Claims 1.1 – 1.6.</p> <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include . . . an image magnifying utility . . . The image magnifying utility will magnify a rectangle of fixed size around a point at which the appropriate gesture was made. The utility allows very accurate positioning of a cursor in the expanded image. After stylus liftoff, the normal size display is restored, and the selected cursor coordinates are sent to the active application program.” Col. 8, ll. 14-36.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to hide the input panel upon user command or upon the opening of another input window. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p> <p>In addition, the combination of Greanias ‘384 and U.S. Patent No. 5,148,155 (“Martin ‘155”) renders this claim obvious. Martin ‘155 teaches the hiding of an input panel. See Martin ‘155 Abstract and Col. 17, ll. 11-22. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem, disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
9	9. The method of claim 1 further comprising, docking the input panel.	<p>See Claim 25 of Greanias ‘384 (“The computer system as recited in claim 14, which further comprises a touch input device disposed over the viewing surface of the display device, and wherein the advanced user interface further comprises a pop-up keyboard module which displays an image of a keyboard on the display device[.]”) (emphasis added).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p> <p>In addition, the combination of Greanias ‘384 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
10	10. At least one computer-readable medium having computer-executable instructions, which when executed perform the method of claim 1.	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 5, ll. 52-63.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
11	11. At least one computer-readable medium having computer-executable instructions stored thereon, which when executed by a computer system perform steps, comprising:	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 5, ll. 52-63.
11.1	selecting one of a plurality of executable input methods for supplying user input to the computer system, wherein each executable input method is an interchangeable software component distinct from one or more application programs, each executable input method having a defined interface set such that	See Claims 1.1 – 1.6. The invention provides for multiple input types which supply user input to the computer system. See Figure 3. The input methods are each separately executable from the application programs. Col. 7, ll. 25 (“the AUI is written as an application program”). The input methods are connectable to the application programs through an advanced user interface and/or the operating system. Col. 1, ll. 5-11 (this invention “relates to an

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
	the executable input method is connectable to the application programs;	advanced user interface which allows a user to select one or more input devices to input data into a computer running a program originally written for a different input device in a multiapplication environment”). See also Figure 3. “The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.” Abstract.
11.2	opening an input window on a display of the computer system independent of a window of an active application program; and	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23. The AUI Utility is independent of the application program. Col. 8, ll. 8-13 (“the AUI utility can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment to communicate with other application programs”).
11.3	displaying an interactive input panel in the input window, the interactive input panel corresponding to the selected executable input method such that information corresponding to user input received by the selected executable input method via the interactive input panel is provided to the active application program as if the information was received via user input at a hardware input device.	See Claims 1.1 – 1.6. “A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i> , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 14-23 (emphasis added). The input methods are each separately executable from the application programs. Col. 7, ll. 25 (“the AUI is written as an application program”). “After receiving an input message from Presentation Manager.TM. 156 together with information about which is the active application program, the PM-Link 201 refers back to the alternate input subsystem 203 to determine whether the “keyboard” or “mouse” message it received is in fact a keyboard or a mouse message, or rather a

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		touch, voice, image, or other message. The PM-Link 201 then passes the true message to the appropriate application program.” Col. 13, ll. 16-25.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
13	13. The computer-readable medium of claim 11 further comprising, providing a Software Input Panel (SIP) menu button on the display of the computer system, the SIP menu button being actuatable to display a selectable list of the plurality of executable input methods.	<p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract.</p> <p>“It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 30-32.</p> <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i>, the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 14-23 (emphasis added).</p> <p>The input methods are each separately executable from the application programs. Col. 7, ll. 25 (“the AUI is written as an application program”).</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
14	14. The computer-readable medium of claim 13 further comprising, receiving a selection of one of the plurality of executable input methods displayed in the list as a selected executable input method, and in response, closing any open input window, and opening a new input window corresponding to the selected executable input method.	See Claim 8. “A few examples of utilities which might be found in the advanced user interface utilities 109 include . . . an image magnifying utility . . . The image magnifying utility will magnify a rectangle of fixed size around a point at which the appropriate gesture was made. The utility allows very accurate positioning of a cursor in the expanded image. After stylus liftoff, the normal size display is restored, and the selected cursor coordinates are sent to the active application program.” Col. 8, ll. 14-36.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
15	15. At least one computer-readable medium having computer-executable instructions, which when executed perform steps, comprising:	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 5, ll. 52-63.
15.1	presenting icons corresponding to a plurality of input methods available for a computer application, wherein each input method is a computer-executable software component distinct from the computer application;	See Claims 1.1 – 1.6. “A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i> , the keyboard or keypad will “pop-up” on the

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120 screen.” Col. 8, ll. 14-23 (emphasis added).
		<p>The AUI Utility is independent of the application program. Col. 8, ll. 8-13 (“the AUI utility can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment to communicate with other application programs”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement an actuatable icon representative of an input method list. For example, the invention discloses several types of interface utilities 109, and it would have been obvious to one of skill in the art to include this step to permit the user to select one of those input utilities. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p>
15.2	invoking a selected input method in response to a user selecting an icon corresponding to the selected input method, including presenting an input panel window; and	<p>See Claims 1.1 – 1.6.</p> <p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.” Col. 8, ll. 20-27 (emphasis added).</p>
15.3	accepting user data entered in the input panel window for the computer application, wherein the user data is provided to the computer application as if the user data was received from a hardware input device.	See Claims 15.2.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
16	16. The computer-readable medium of claim 15 wherein accepting user data includes detecting user interaction with a touch-sensitive display.	<p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i>, the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.” Col. 8, ll. 20-27 (emphasis added).</p> <p>See Claim 17 of Greanias ‘384 (“The computer system as recited in claim 14, which further comprises a gesture recognition unit which interprets the input signals transmitted by a touch input device in response to a user drawing symbols on a surface detected by the touch input device.”).</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
17	17. The computer-readable medium of claim 15 wherein each input method comprises a component object model (COM) object, and wherein the step of invoking the selected input method includes the step of instantiating the COM object.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 3, ll. 13-27; Col. 8, ll. 13-27. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Greanias ‘384 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		<p>solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
18	18. The computer-readable medium of claim 15 further comprising converting the user data to a Unicode character value.	<p>“The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.” Abstract.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convert the user data to a Unicode character value. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		<p>In addition, the combination of Greanias ‘384 and U.S. Patent No. 5,454,046 (“the Carman Reference”)⁴ renders this claim obvious. See Carman Reference, Col. 10, ll. 14-22 (“The text represented at block 62 may consist of one or more words represented in Unicode, a 16-bit character representation defined by the Unicode Consortium of Mountain View, Calif. This representation is used instead of ASCII or ANSI representations, since Unicode can accommodate over 65,000 character representations, representing the definition of character glyphs of virtually all languages of the world, whereas ASCII and ANSI representations are limited to 256 characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>See Figure 3A of Carman Reference.</p>

⁴ “Universal Symbolic Handwriting Recognition System” issued to Carman; filed September 17, 1993; issued September 26, 1995.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		<div data-bbox="285 478 721 987"> </div> <p data-bbox="755 693 792 819">FIG. 3A</p> <p data-bbox="836 210 974 1281">Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p data-bbox="998 168 1315 1281">Similarly, the combination of Greanias ‘384 and U.S. Patent No. 5,455,901 (“the Friend Reference”)⁵ renders this claim obvious. See Claim 6 of Friend Reference (“The device of claim 5 wherein said translator comprises a generator for generating machine readable characters from said strokes.”); Claim 7 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise Unicode characters.”); Claim 8 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise ASCII characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an</p>

⁵ “Input Device With Deferred Translation” issued to Friend; filed September 12, 1994; issued October 3, 1995.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 6,339,780	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
19	19. In a computing environment, a system comprising,	“An advanced user interface for use with a computer system operating on an integrated operating environment.” Abstract.
19.1	a manager component stored on one or more computer-readable media and configured: to manage selection of a selected input method from one or more available stored input methods, wherein each input method is a computer-executable software component distinct from one or more computer programs, and to send input data corresponding to a user input received at the selected input method to a graphical windowing	“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract. “It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 30-32. The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The input methods are each separately executable from the application programs. Col. 7, ll. 25 (“the AUI is written as an application program”); Col. 8, ll. 8-13 (“the AUI utility can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment to communicate with other application programs”). An example input

Claim	U.S. Patent 6,339,780	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
	environment; and	method application is a “pop-up keyboard.” Col. 8, ll. 5-8 (“The Advanced User Interface utility (AUI utility) 109 is a set of utility programs such as a pop-up keyboard or an image magnifier utility which can be used with AUI 100.”). The AUI has “translation capabilities” for sending user data to the application programs. Col. 7, ll. 15.
19.2	the graphical windowing environment to receive the input data and to send the input data to a computer program of the one or more computer programs, wherein the input data is sent to the computer program as if the input data was received via user input received from a hardware input device.	The Integrated Operating Environment (windowing environment) serves as the interface between the AUI and the applications. See Figures 3 & 6; see also Abstract (“The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.”). The AUI Utility is independent of the application program. Col. 8, ll. 8-13 (“the AUI utility can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment to communicate with other application programs”).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
20	20. The system of claim 19 wherein the computer program comprises an application program having focus.	“The integrated operating environment allows a plurality of application programs to be running simultaneously, one of which is designated the active application program to which all input data is directed.” Abstract.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
21	21. The system of claim 19 further comprising an input panel window corresponding to the selected input method.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
22	22. The system of claim 21 wherein the selected input method presents an image representing a keyboard on the input panel window.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
23	23. The system of claim 21 wherein the manager component selectively displays and hides the input panel window.	See Claim 8. “By issuing the proper command, e.g., a circle gesture or touching a keyboard icon , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
-------	-----------------------	--

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
25	25. The system of claim 19 where the input method is displayed on a touch-sensitive display screen.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i> , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
26	26. The system of claim 19 wherein the manager component transfers information from the computer program to the selected input method.	See Figure 8. “After being recognized by the gesture recognition unit 213 as a circle gesture, the PM-Link 201 passes the circle gesture to the spreadsheet 150. The spreadsheet returns an “R0” message at 285, indicating that the circle gesture was not understood. Since the circle gesture was not understood at 287, the PM-Link 201 refers to the application profile 205 for the spreadsheet 150 at 289 and finds no corresponding command for the circle gesture. The PM-Link 201 then refers to the user profile 207 at 291 which contains the corresponding command “invoke the pop-up keyboard”. As the command from the user profile 207 has the highest priority at 293, the PM-Link 201 sends this message to the AUI utilities module 209 at 295 and the pop-up keyboard is presented to the user on the display.” Col. 16, ll. 17-31.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
27	27. The system of claim 19 wherein the selected input method calls functions in the manager component via a defined	See Figure 4. See Claim 37 of Greanias ‘384 (“The set of interface profiles as recited in claim 31, wherein the interface profiles contain mappings of touch input messages against

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
	interface set.	corresponding mouse commands.”).
		“After determining the active application program, spreadsheet 50 at 119, the environment link 101 refers to the application profile 105 of the spreadsheet 50 for the command which corresponds to the input message “GRAPH” which will be recognized by the spreadsheet 50, i.e., the menu selection-- graph--.” Col. 8, ll. 59-65.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
28	28. The system of claim 19 wherein the selected input method comprises an object.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 3, ll. 13-27; Col. 8, ll. 13-27. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Greanias ‘384 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
29	29. The system of claim 19 wherein the selected input method draws an input panel in an input panel window displayed in the graphical windowing environment.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
30	30. The system of claim 29 wherein the manager component selectively displays and hides the display of the input panel	See Claim 8. “By issuing the proper command, e.g., a circle gesture or touching a keyboard icon , the keyboard or keypad will “pop-up” on the screen.” Col. 8, ll. 20-23 (emphasis).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
	window.	

Claim	U.S. Patent 7,411,582	U.S. Patent 5,157,384 (“Greanias ‘384’”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
31	31. The system of claim 29 wherein the manager component docks the input panel window.	<p>See Claim 9.</p> <p>See Claim 25 of Greanias ‘384 (“The computer system as recited in claim 14, which further comprises a touch input device disposed over the viewing surface of the display device, and wherein the advanced user interface further comprises a pop-up keyboard module which displays an image of a keyboard on the display device[.]”) (emphasis added).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 3, ll. 13-27; Col. 8, ll. 13-27.</p> <p>In addition, the combination of Greanias ‘384 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

EXHIBIT I

Exemplar Chart of U.S. Patent 7,411,582**U.S. Patent 5,148,155 (“Martin ‘155”)¹
Claims 1-4, 6, 8-11, 13-23, 25-31**

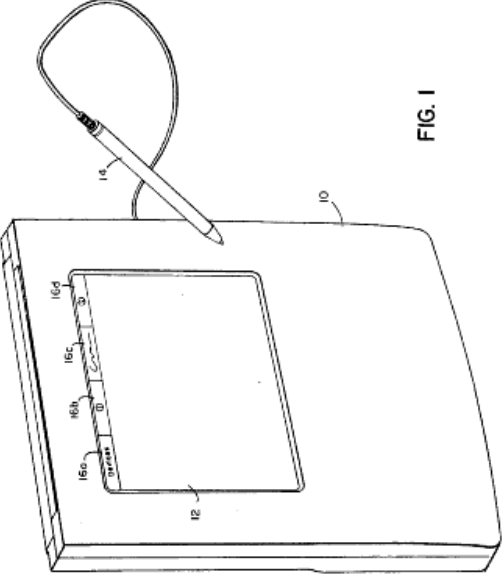
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt ² and/or the MessagePad 120 ³
1	1. In a computing environment, a computer-implemented method comprising:	“A computer system having a digitizing tablet overlaying the display screen.” Abstract.
1.1	displaying an actuatable icon representative of an input method list that includes one or more selectable input methods for one or more computer programs, wherein each input method is a computer-executable software component distinct from the computer	The DEVICES button is an actuatable icon. See Figure 1.

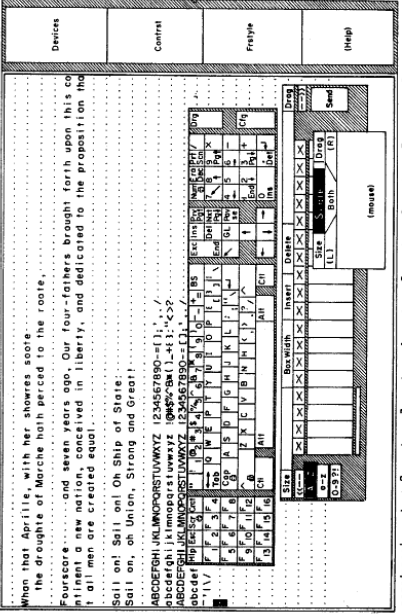
NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Computer With Tablet Input To Standard Programs” issued to Martin et al.; filed November 13, 1990; issued September 15, 1992.

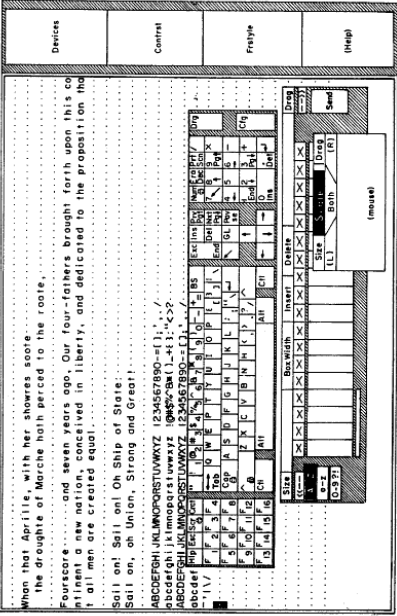
² Kraig Brockschmidt, Inside OLE (2d ed. 1995).

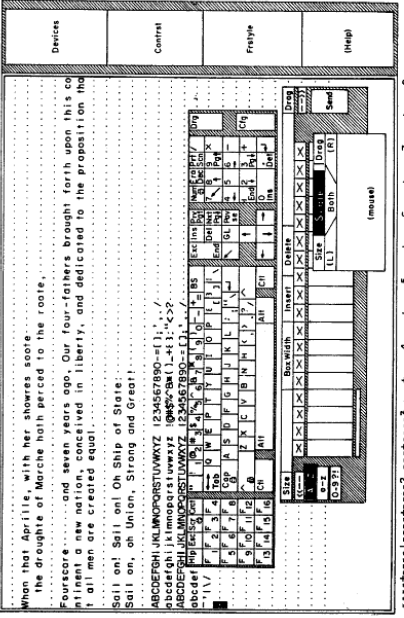
³ Upon information and belief, the MessagePad 120 was first sold in the United States in January of 1995. The Newton Apple MessagePad Handbook would have been distributed to the public around the same time.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt ² and/or the MessagePad 120 ³
	programs;	 <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed.” Col. 17, ll. 11-15.</p> <p>“[T]hese devices are independent of the various application programs with which they may be used. In the preferred embodiment, they are even implemented in part using a processor separate from the processor on which the application is running.” Col. 16, ll. 30-35.</p>
1.2	in response to actuation of the actuatable icon, displaying the input method list;	<p>See Claim 1.1 regarding the actuation of the actuatable icon.</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed.” Col. 17, ll. 11-15.</p> <p>See Figure 10.</p>

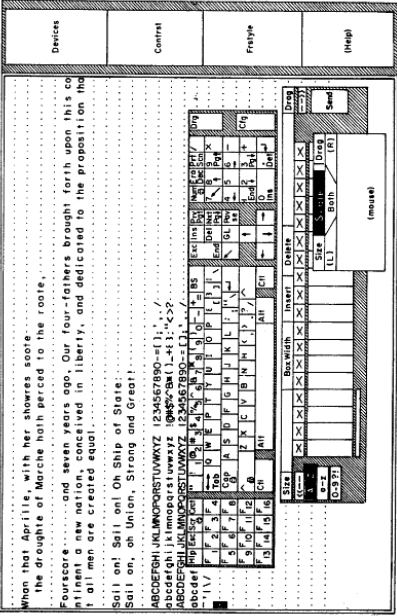
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt ² and/or the MessagePad 120 ³
1.3	receiving a selection of an input method from the input method list;	 <p>FIG. 10</p>
		<p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active[.]” Col. 17, ll. 11-16.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt ² and/or the MessagePad 120 ³
1.4	installing an input method component that corresponds to the selected input method, the input method component causing an interactive input panel to be displayed;	<p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30.</p> <p>Once installed, the input method may communicate with the active application. Col. 18, ll. 12-16 (“FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data.”).</p>
1.5	receiving input via the interactive input panel; and	See Figure 10.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt ² and/or the MessagePad 120 ³
		 <p>FIG. 10</p> <p>"The simulated keyboard device allows users to enter keyboard data by 'typing' with the stylus on an iconic keyboard." Col. 27, ll. 54-56.</p> <p>"The primary body of the icon for the handwriting recognition input device contains boxes into which the user can write characters." Col. 21, ll. 46-48.</p>
1.6	providing the input to a computer program of the one or more computer programs as if the information was received via user input received from a hardware input device.	<p>"Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor's BIOS 212 and services provided by the interface processor's firmware 214." Col. 34, ll. 8-15.</p> <p>"FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data." Col. 18, ll. 12-16.</p> <p>See Figure 10.</p>

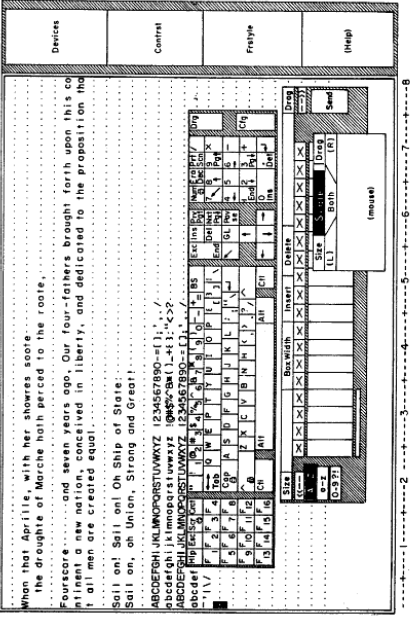
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt ² and/or the MessagePad 120 ³
		

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
2	2. The method of claim 1 wherein providing the input to the computer program comprises communicating information representative of the input to a graphical windowing environment.	<p>"Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor's BIOS 212 and services provided by the interface processor's firmware 214." Col. 34, ll. 8-15.</p> <p>"FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data." Col. 18, ll. 12-16.</p> <p>See Figure 10.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		 <p>FIG. 10</p>
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
3	3. The method of claim 2 wherein communicating the information comprises passing the information to an interface.	<p>"In order to effect this overlaying <i>interface</i> without interfering with the execution of the pre-existing program, an interface processor is provided to execute software used in implementing this interface." Col. 3, ll. 37-40 (emphasis added).</p> <p>"Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor's BIOS 212 and services provided by the interface processor's firmware 214." Col. 34, ll. 8-15.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
4	4. The method of claim 2 further comprising, communicating the information from the graphical windowing environment to an application program, wherein the computer program includes the application program, wherein the information is provided to the application program in a same manner as if the input was received via a hardware keyboard.	<p>“Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor’s BIOS 212 and services provided by the interface processor’s firmware 214.” Col. 34, ll. 8-15.</p> <p>“The interface processor manages input from the tablet . . . and provides keystroke and mouse data to the main processor as if from a standard keyboard controller.” Abstract.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
6	6. The method of claim 1 wherein the selected input method corresponds to a displayed keyboard, and wherein receiving input via the interactive input panel that corresponds to the selected input method comprises receiving information corresponding to a keyboard character entered via the displayed keyboard.	<p>“The simulated keyboard device allows users to enter keyboard data by ‘typing’ with the stylus on an iconic keyboard.” Col. 27, ll. 54-56.</p> <p>See Figure 10.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		 <p>FIG. 10</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
8	8. The method of claim 1 further comprising, hiding the input panel.	<p>"During interaction with one of the user's programs, the user can activate and deactivate simulated devices (by removing them from and returning them to a device tray)[.]" Abstract.</p> <p>"When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user 'operates'." Col. 17, ll. 11-30.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
-------	-----------------------	--

9. The method of claim 1 further comprising, docking the input panel.

See Figure 10.

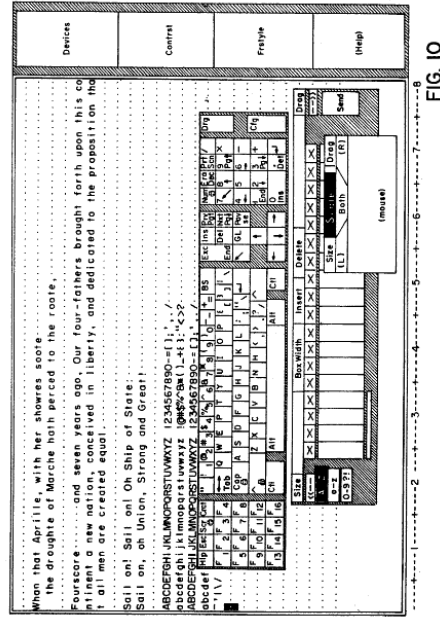


FIG. 10

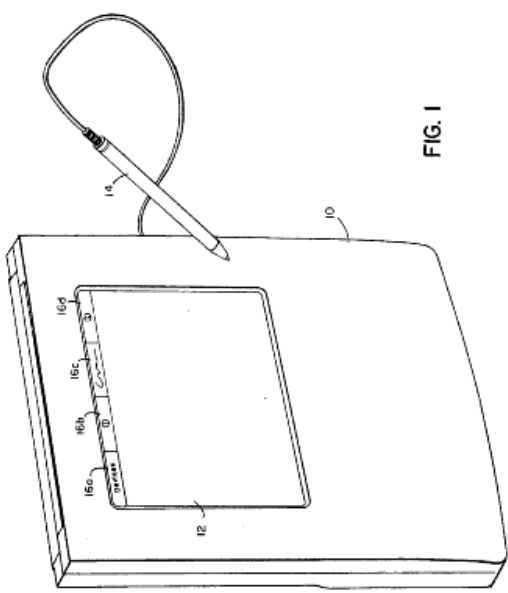
To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 55, ll. 55-68 to Col. 56, ll. 55-56.

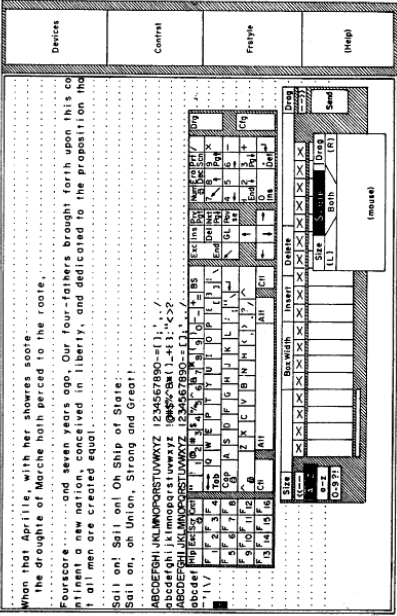
In addition, the combination of Martin '155 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field

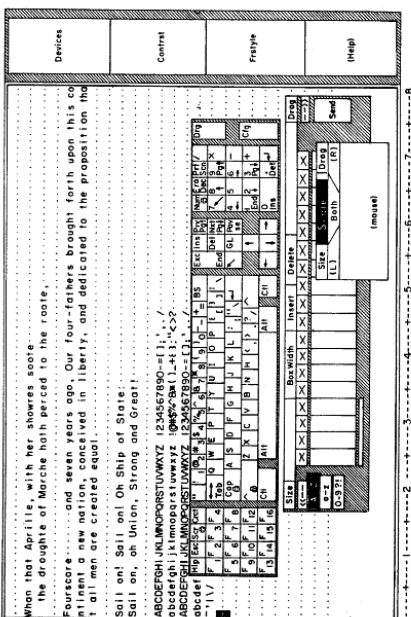
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
10	10. At least one computer-readable medium having computer-executable instructions, which when executed perform the method of claim 1.	“In order to effect this overlaying interface without interfering with the execution of the pre-existing program, an interface processor is provided to execute software used in implementing this interface.” Col. 3, ll. 37-40.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
11	11. At least one computer-readable medium having computer-executable instructions stored thereon, which when executed by a computer system perform steps, comprising:	“In order to effect this overlaying interface without interfering with the execution of the pre-existing program, an interface processor is provided to execute software used in implementing this interface.” Col. 3, ll. 37-40.

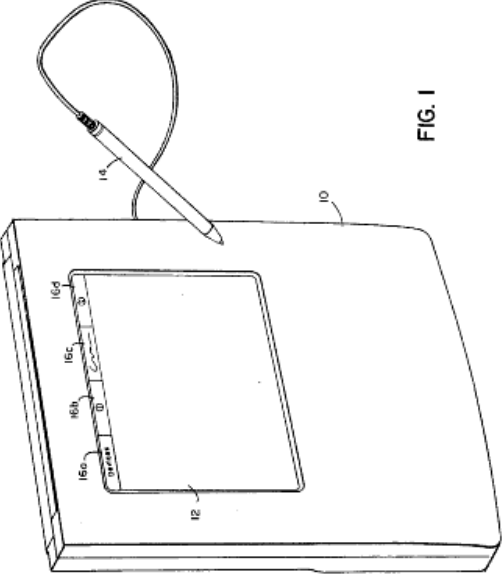
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
11.1	selecting one of a plurality of executable input methods for supplying user input to the computer system, wherein each executable input method is an interchangeable software component distinct from one or more application programs, each executable input method having a defined interface set such that the executable input method is connectable to the application programs;	<p>See Figure 1.</p>  <p>FIG. 1</p> <p>See Figure 10.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		 <p>FIG. 10</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30.</p> <p>“[T]hese devices are independent of the various application programs with which they may be used. In the preferred embodiment, they are even implemented in part using a processor separate from the processor on which the application is running.” Col. 16, ll. 30-35.</p>
11.2	opening an input window on a display of	“FIG. 10 is a screen display showing the icons for three simulated devices. Behind

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120	these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data.” Col. 18, ll. 12-16. See Figure 10.
	the computer system independent of a window of an active application program; and	 FIG. 10	these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data.” Col. 18, ll. 12-16. See Figure 10.
11.3	displaying an interactive input panel in the input window, the interactive input panel corresponding to the selected executable input method such that information corresponding to user input received by the selected executable input method via the interactive input panel is provided to the active application program as if the information was received via user input at a hardware input device.		“FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data.” Col. 18, ll. 12-16. See Figure 10.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		<p style="text-align: center;">FIG. 10</p>
		"The interface processor manages input from the tablet . . . and provides keystroke and mouse data to the main processor as if from a standard keyboard controller." Abstract.

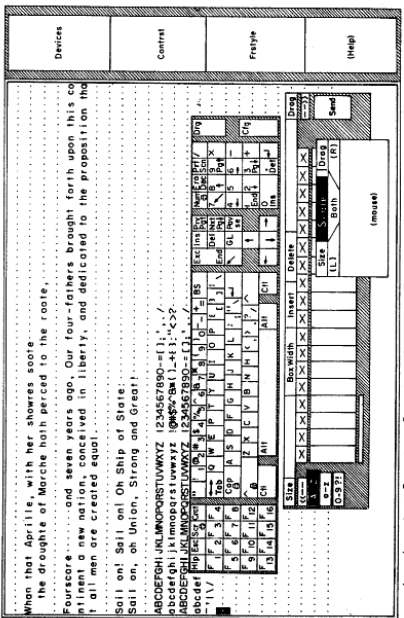
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
13	13. The computer-readable medium of claim 11 further comprising, providing a Software Input Panel (SIP) menu button on the display of the computer system, the SIP menu button being actuable to display a selectable list of the plurality of executable input methods.	The DEVICES button is the SIP menu button. See Figure 1.

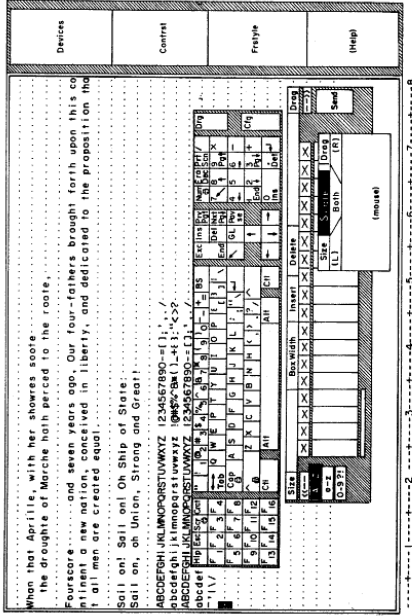
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		 <p>FIG. 1</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed.” Col. 17, ll. 11-15.</p>

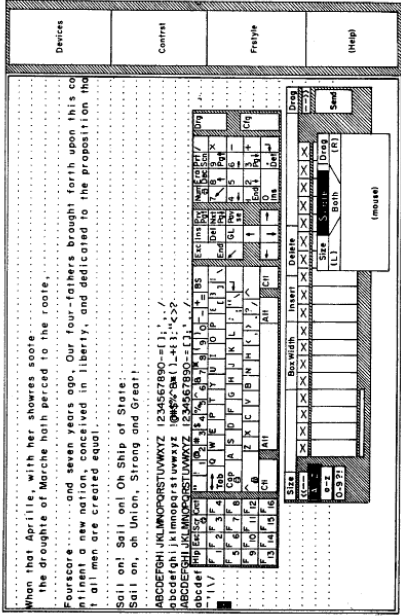
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
14	14. The computer-readable medium of claim 13 further comprising, receiving a selection of one of the plurality of executable input methods displayed in the list as a selected executable input	<p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
	method, and in response, closing any open input window, and opening a new input window corresponding to the selected executable input method.	results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
15	15. At least one computer-readable medium having computer-executable instructions, which when executed perform steps, comprising:	“In order to effect this overlaying interface without interfering with the execution of the pre-existing program, an interface processor is provided to execute software used in implementing this interface.” Col. 3, ll. 37-40.
15.1	presenting icons corresponding to a plurality of input methods available for a computer application, wherein each input method is a computer-executable software component distinct from the computer application;	See Figure 10.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
15.2	invoking a selected input method in response to a user selecting an icon	 <p>FIG. 10</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30.</p> <p>“[T]hese devices are independent of the various application programs with which they may be used. In the preferred embodiment, they are even implemented in part using a processor separate from the processor on which the application is running.” Col. 16, ll. 30-35.</p> <p>See Figure 10.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
corresponding to the selected input method, including presenting an input panel window; and		<p>U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
15.3	accepting user data entered in the input panel window for the computer application, wherein the user data is provided to the computer application as if the user data was received from a hardware input device.	<p>"FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data." Col. 18, ll. 12-16.</p> <p>See Figure 10.</p>  <p>FIG. 10</p> <p>"The interface processor manages input from the tablet . . . and provides keystroke and mouse data to the main processor as if from a standard keyboard controller." Abstract.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
16	16. The computer-readable medium of claim 15 wherein accepting user data includes detecting user interaction with a	<p>See Figure 1.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
	touch-sensitive display.	<div data-bbox="300 438 787 1005" data-label="Image"> </div> <div data-bbox="699 529 727 585" data-label="Caption">FIG. 1</div> <div data-bbox="852 1092 896 1285" data-label="Text">See Figure 10.</div> <div data-bbox="904 426 1317 1039" data-label="Image"> </div> <div data-bbox="1317 464 1344 529" data-label="Caption">FIG. 10</div> <div data-bbox="1385 287 1466 1285" data-label="Text"> <p>"A computer system having a digitizing tablet overlaying the display screen." Abstract.</p> </div>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
17	17. The computer-readable medium of claim 15 wherein each input method comprises a component object model (COM) object, and wherein the step of invoking the selected input method includes the step of instantiating the COM object.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 55, ll. 55-68 to Col. 56, ll. 55-56. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Martin ‘155 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
18	18. The computer-readable medium of claim 15 further comprising converting the user data to a Unicode character value.	To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convert the user data to a Unicode character value. Col. 55, ll. 55-68 to Col. 56, ll. 55-56. In addition, the combination of Martin ‘155 and U.S. Patent No. 5,454,046 (“the Carman Reference”) ⁴ renders this claim obvious. See Carman Reference, Col. 10, ll. 14-22 (“The text represented at block 62 may consist of one or more words represented in Unicode, a 16-bit character representation defined by the Unicode Consortium of Mountain View, Calif. This representation is used instead of ASCII or ANSI representations, since Unicode can accommodate over 65,000 character representations, representing the definition of character glyphs of virtually all languages of the world, whereas ASCII and ANSI representations are limited to 256 characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem

⁴ “Universal Symbolic Handwriting Recognition System” issued to Carman; filed September 17, 1993; issued September 26, 1995.

Claim	U.S. Patent 7,411,582	<p>U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120</p>
		<p>(permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>See Figure 3A of Carman Reference.</p> <div data-bbox="483 478 912 987" data-label="Diagram"> <pre> graph TD 48[USER SPECIFIC SAMPLE FILE OF TEXT / PATTERN PAIRS] 62[UNICODE TEXT] 64[PATTERN] 66[INDEX] 68[FEATURES] 70[DICTIONARIES] DBS[DATA BASE STORAGE] 48 --- 62 48 --- 64 48 --- 66 64 --- 68 48 --- 70 48 --- DBS </pre> </div> <p>FIG. 3A</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>Similarly, the combination of Martin ‘155 and U.S. Patent No. 5,455,901 (“the Friend Reference”)⁵ renders this claim obvious. See Claim 6 of Friend Reference (“The device of claim 5 wherein said translator comprises a generator for generating</p>

⁵ “Input Device With Deferred Translation” issued to Friend; filed September 12, 1994; issued October 3, 1995.

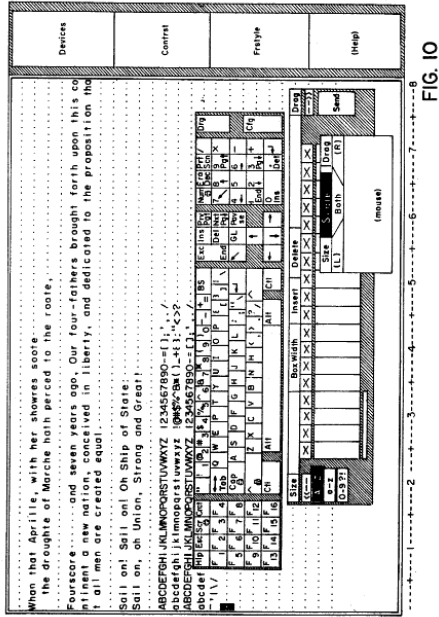
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		machine readable characters from said strokes.”); Claim 7 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise Unicode characters.”); Claim 8 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise ASCII characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 6,339,780	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
19	19. In a computing environment, a system comprising,	“A computer system having a digitizing tablet overlaying the display screen.” Abstract.
19.1	a manager component stored on one or more computer-readable media and configured: to manage selection of a selected input method from one or more available stored input methods, wherein each input method is a computer-executable	See Figure 2.

Claim	U.S. Patent 6,339,780	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
	<p>software component distinct from one or more computer programs, and</p> <p>to send input data corresponding to a user input received at the selected input method to a graphical windowing environment; and</p>	<p style="text-align: center;">FIG. 2</p> <p>See Figure 10.</p> <p style="text-align: right;">FIG. 10</p> <p>See Figure 11.</p>

Claim	U.S. Patent 6,339,780	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		<p style="text-align: center;">FIG. 11</p> <p>“The interface processor <i>manages</i> input from the tablet . . . and provides keystroke and mouse data to the main processor as if from a standard keyboard controller.” Abstract (emphasis added).</p> <p>“Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor’s BIOS 212 and services provided by the interface processor’s firmware 214.” Col. 34, ll. 8-15.</p> <p>“FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data.” Col. 18, ll. 12-16.</p> <p>See Figure 10.</p>

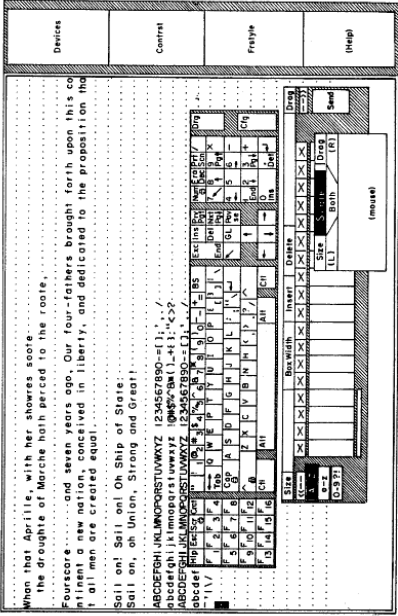
Claim	U.S. Patent 6,339,780	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
19.2	the graphical windowing environment to receive the input data and to send the input data to a computer program of the one or more computer programs, wherein the input data is sent to the computer program as if the input data was received via user input received from a hardware input device.	<p>See Figures 2, 10 & 11.</p> <p>"The interface processor <i>manages</i> input from the tablet . . . and provides keystroke and mouse data to the main processor as if from a standard keyboard controller." Abstract (emphasis added).</p> <p>"Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor's BIOS 212 and services provided by the interface processor's firmware 214." Col. 34, ll. 8-15.</p> <p>"FIG. 10 is a screen display showing the icons for three simulated devices. Behind these icons is the display of a text editing application adapted to receive input in the form of keystrokes and mouse data, and not adapted to receive tablet data." Col. 18, ll. 12-16.</p>

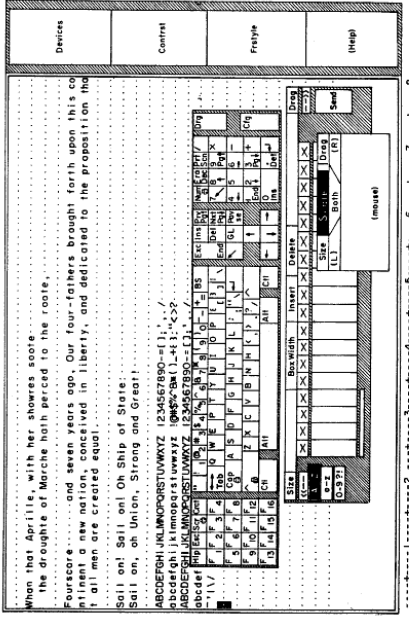


Claim	U.S. Patent 6,339,780	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
20	20. The system of claim 19 wherein the computer program comprises an application program having focus.	<p>See Figure 11.</p> <p style="text-align: center;">FIG. 11</p> <p>“To receive data from the interface processor, an application must provide a routine that is to be called when data is received from the interface processor. The application registers this data handler routine with the datalink driver by the ‘install’ BIOS call mentioned above.” Col. 14, ll. 63-68.</p> <p>“The BIOS for the main processor provides a mechanism by which an application running in the main processor can register to receive tablet data.” Col. 15, ll. 62-64.</p> <p>“Although block 200 is identified as “Application Program”, it represents any program (including system programs) with which a user may wish to interact via the</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		tablet and display." Col. 33, ll. 32-35.
		To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system to send user input to an application program having focus. Col. 55, ll. 55-68 to Col. 56, ll. 55-56.

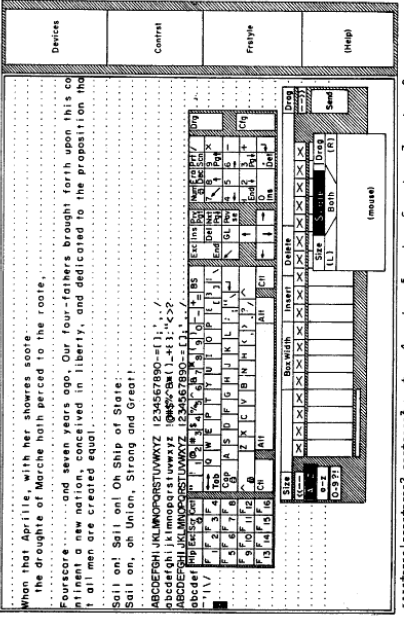
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
21	21. The system of claim 19 further comprising an input panel window corresponding to the selected input method.	See Figure 10.  FIG. 10 "The simulated keyboard device allows users to enter keyboard data by "typing" with the stylus on an iconic keyboard." Col. 27, ll. 54-56.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
22	22. The system of claim 21 wherein the selected input method presents an image representing a keyboard on the input panel window.	See Figure 10. 
		"The simulated keyboard device allows users to enter keyboard data by "typing" with the stylus on an iconic keyboard." Col. 27, ll. 54-56.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
23	23. The system of claim 21 wherein the manager component selectively displays and hides the input panel window.	"During interaction with one of the user's programs, the user can activate and deactivate simulated devices (by removing them from and returning them to a device tray)[.]" Abstract. "When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is displayed. The user can then drag from the tray those devices that the user wishes to

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates.’” Col. 17, ll. 11-30.
		“Provisions can be made for an application to invoke user services provided by the user interface subsystem. An application running in the main processor can make a particular call, which causes the main processor to pass a code to the interface processor, identifying the service to be performed. For example, it is desirable for certain applications to be able to force certain simulated devices to appear and disappear at appropriate times.” Col. 32, ll. 36-44.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
25	25. The system of claim 19 where the input method is displayed on a touch-sensitive display screen.	See Figure 10.

Claim	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
	 <p>FIG. 10</p> <p>“The simulated keyboard device allows users to enter keyboard data by “typing” with the stylus on an iconic keyboard.” Col. 27, ll. 54-56.</p> <p>“A computer system having a digitizing tablet overlaying the display screen.” Abstract.</p>

Claim	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
26	<p>26. The system of claim 19 wherein the manager component transfers information from the computer program to the selected input method.</p> <p>“Provisions can be made for an application to invoke user services provided by the user interface subsystem. An application running in the main processor can make a particular call, which causes the main processor to pass a code to the interface processor, identifying the service to be performed. For example, it is desirable for certain applications to be able to force certain simulated devices to appear and disappear at appropriate times.” Col. 32, ll. 36-44.</p>

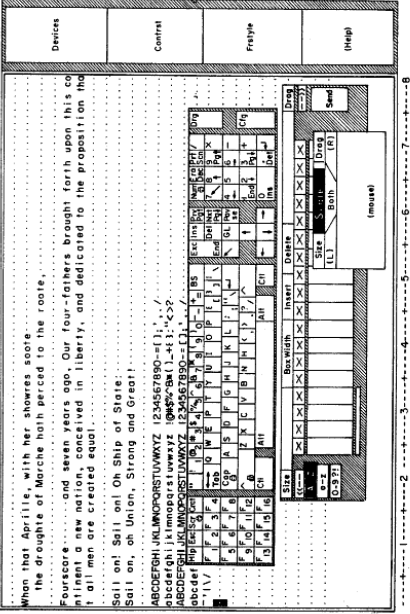
Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
27	27. The system of claim 19 wherein the selected input method calls functions in the manager component via a defined interface set.	<p>“In order to effect this overlaying <i>interface</i> without interfering with the execution of the pre-existing program, an interface processor is provided to execute software used in implementing this interface.” Col. 3, ll. 37-40 (emphasis added).</p> <p>“Data from the tablet 12b is processed by the simulated devices program 220, which, in response thereto, generates data, typically in the form of keystrokes or mouse data. This data is provided to the application program 200, by means of communication services 210 that include services provided by the main processor’s BIOS 212 and services provided by the interface processor’s firmware 214.” Col. 34, ll. 8-15.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
28	28. The system of claim 19 wherein the selected input method comprises an object.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 55, ll. 55-68 to Col. 56, ll. 55-56. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		<p>rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Martin ‘155 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
29	29. The system of claim 19 wherein the selected input method draws an input panel in an input panel window displayed	See Figure 10.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
	in the graphical windowing environment.	 <p>FIG. 10</p> <p>“The simulated keyboard device allows users to enter keyboard data by “typing” with the stylus on an iconic keyboard.” Col. 27, ll. 54-56.</p> <p>“The primary body of the icon for the handwriting recognition input device contains boxes into which the user can write characters.” Col. 21, ll. 46-48.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
30	30. The system of claim 29 wherein the manager component selectively displays and hides the display of the input panel window.	<p>“During interaction with one of the user’s programs, the user can activate and deactivate simulated devices (by removing them from and returning them to a device tray)].” Abstract.</p> <p>“When the user initially presses the DEVICES button (by touching that portion of the tablet corresponding to the device button), a tray of the available simulated devices is</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		displayed. The user can then drag from the tray those devices that the user wishes to be active, and can drag onto the tray those devices previously activated, but which the user does not presently need. The user then presses the DEVICES button again, which results in removal (from the display) of the tray and any devices on it. For convenience, the devices appear on the tray as small icons (referred to as stamps); also, when the tray is displayed, devices present on the screen because they were previously removed from the tray are also displayed in stamp form; when the tray is removed, the visual representation of any of the remaining devices (i.e., not located on the tray) are replaced by the larger, typically more complex, icons which are the visual forms of the simulated devices that the user ‘operates’.” Col. 17, ll. 11-30.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 ("Martin '155") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
31	31. The system of claim 29 wherein the manager component docks the input panel window.	See Figure 10.

FIG. 10

Claim	U.S. Patent 7,411,582	U.S. Patent 5,148,155 (“Martin ‘155”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,157,384 and/or Brockschmidt and/or the MessagePad 120
		<p>“As with the other device icons, the keyboard icon includes a drag sub-icon, by which the user can position the keyboard icon on the display.” Col. 29, ll. 1-4.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 55, ll. 55-68 to Col. 56, ll. 55-56.</p> <p>In addition, the combination of Martin ‘155 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

EXHIBIT J

Exemplar Chart of U.S. Patent 7,411,582**Apple Newton MessagePad 120 (“MessagePad 120”)¹
Claims 1-4, 6, 8-11, 13-23, 25-31**

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide ² and/or The NewtonScript Programming Language ³ and/or Brockschmidt ⁴
1	1. In a computing environment, a computer-implemented method comprising:	The MessagePad 120 is a handheld computer device running Newton OS that permits the user to select from among multiple input methods while entering data into an active application. These input methods are designed to work with a plurality of applications. The use of the various input methods of the MessagePad 120 with the Notepad and Calls applications are shown in this chart.
1.1	displaying an actuatable icon representative of an input	The screen cover of the MessagePad 120 describes the keyboard icon as follows:

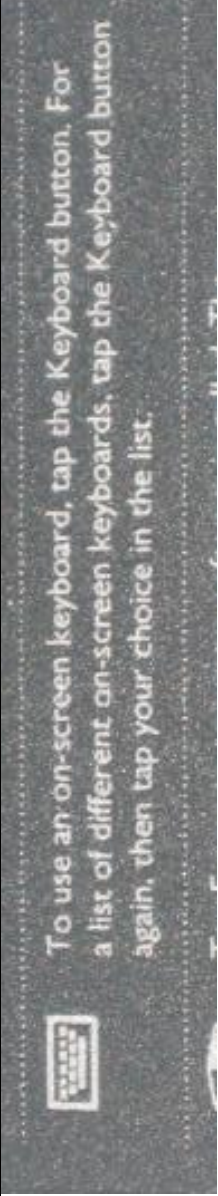
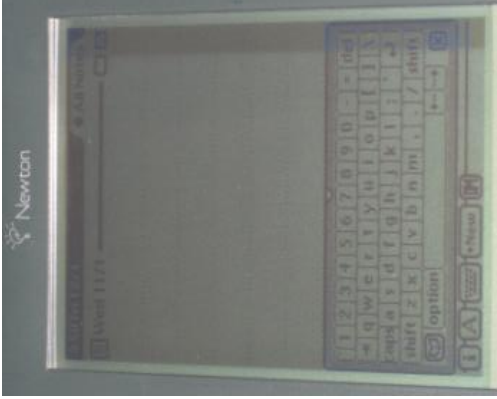
NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

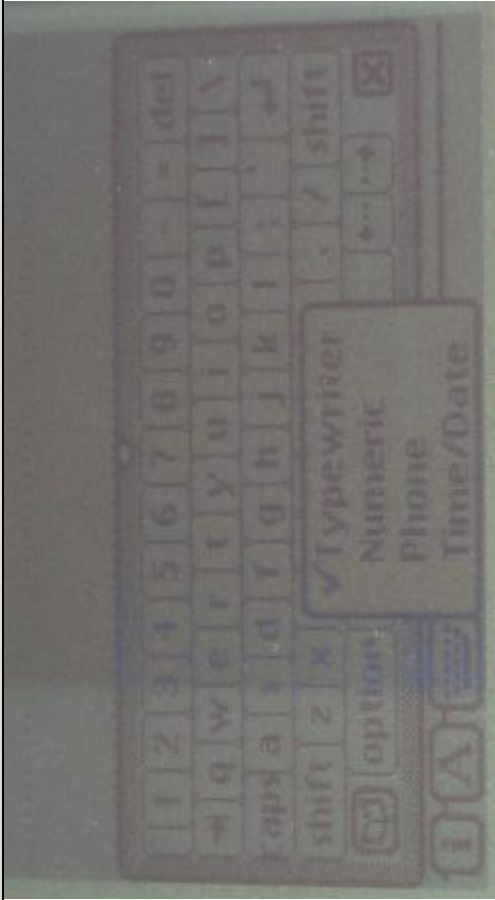
¹ Upon information and belief, the MessagePad 120 was first sold in the United States in January of 1995. The Newton Apple MessagePad Handbook would have been distributed to the public around the same time.

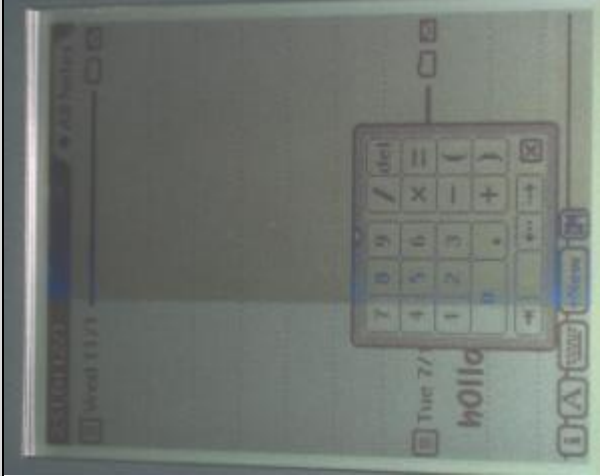
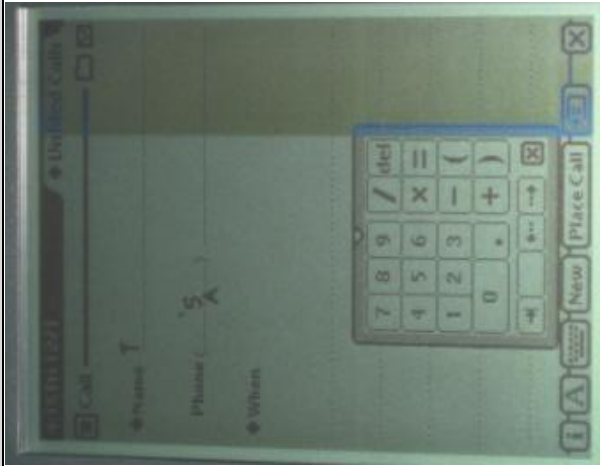
² Apple Press, Apple Computer Inc., Newton’s Programmer’s Guide, For Newton 2.0, Addison Wesley Publishing Company (1996).

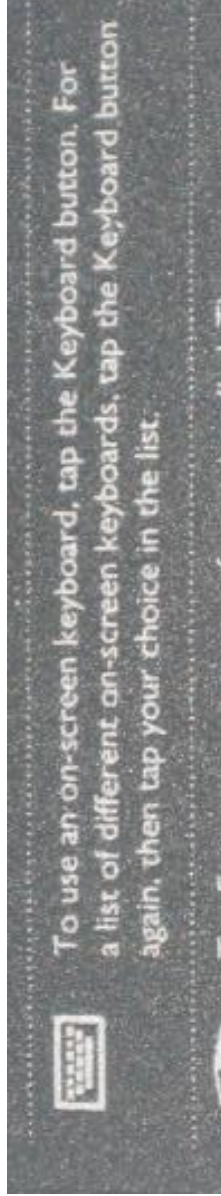
³ Apple Press, Apple Computer Inc., The NewtonScript Programming Language, Addison Wesley Publishing Company (1996).


⁴ Kraig Brockschmidt, Inside OLE (2d ed. 1995).

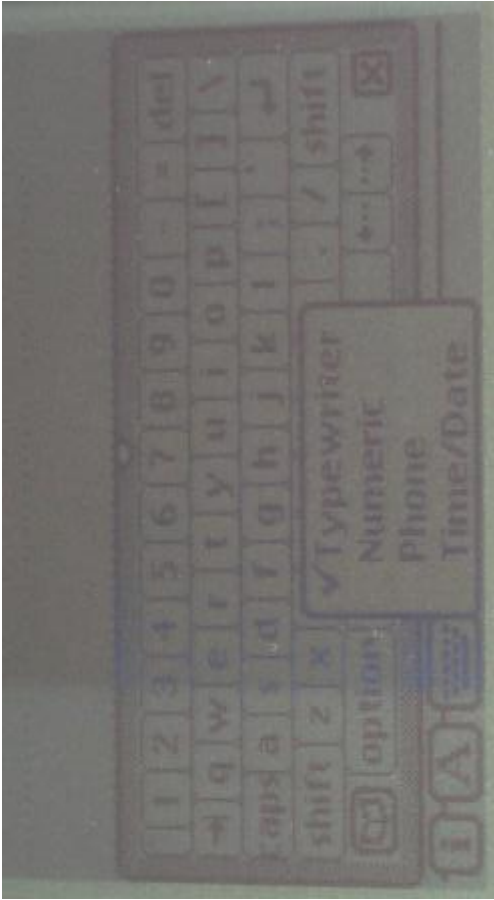
Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴</p>
	<p>method list that includes one or more selectable input methods for one or more computer programs, wherein each input method is a computer-executable software component distinct from the computer programs;</p>	 <p>The MessagePad 120 displays a keyboard icon along the status bar. As shown below, when pressed once, an on-screen keyboard appears; when pressed twice, the other available input methods are shown.</p> 


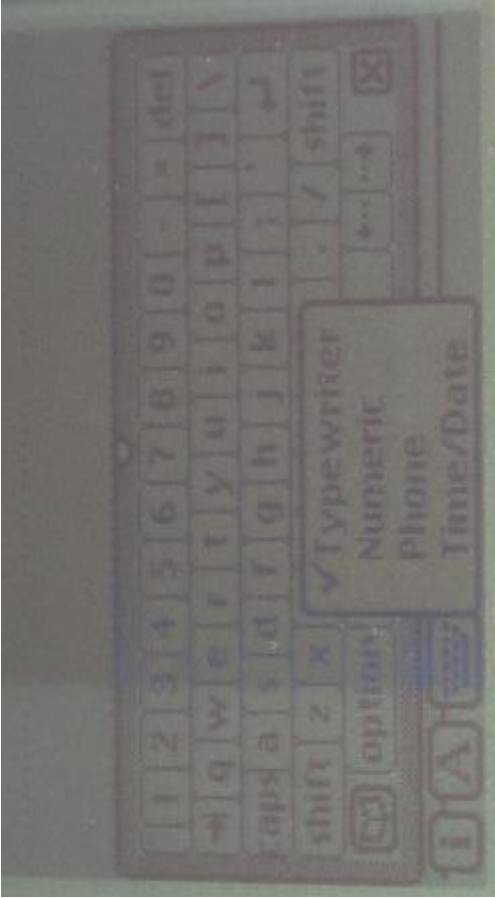
Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴</p>
		 <p>The input methods are distinct from the applications. As shown below, the on-screen Numeric keyboard works with the Notepad and Calls applications.</p>

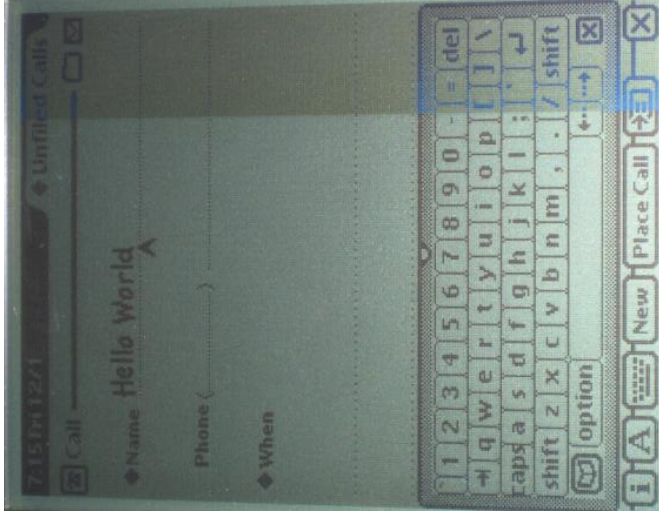
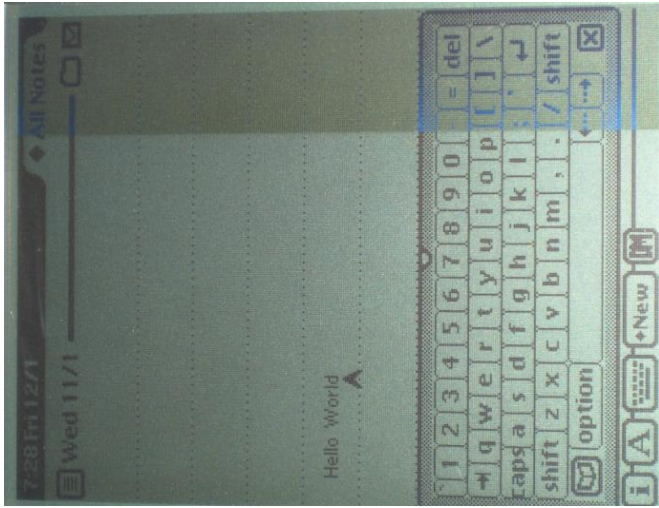
Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴</p>
		<div data-bbox="321 798 917 1270">  </div> <div data-bbox="321 304 917 766">  </div> <p>The Newton Programmer’s Guide further illustrates this capability. See Newton Programmer’s Guide at page 8-26:</p> <div data-bbox="1047 966 1096 1239"> <h3>Using Keyboards</h3> </div> <div data-bbox="1112 346 1209 1239"> <p>You can provide the user with on-screen keyboard input in your applications using the built-in keyboard views. You can also define new keyboard views and register them with the system, which will activate caret input when these views are opened.</p> </div> <div data-bbox="1226 1008 1274 1239"> <h3>Keyboard Views</h3> </div> <div data-bbox="1282 346 1356 1239"> <p>There are four different floating keyboards built into the system root view. Each of the built-in keyboards can be accessed as a child of the root with a symbol.</p> </div> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design an input system capable of working with virtually any application.</p>

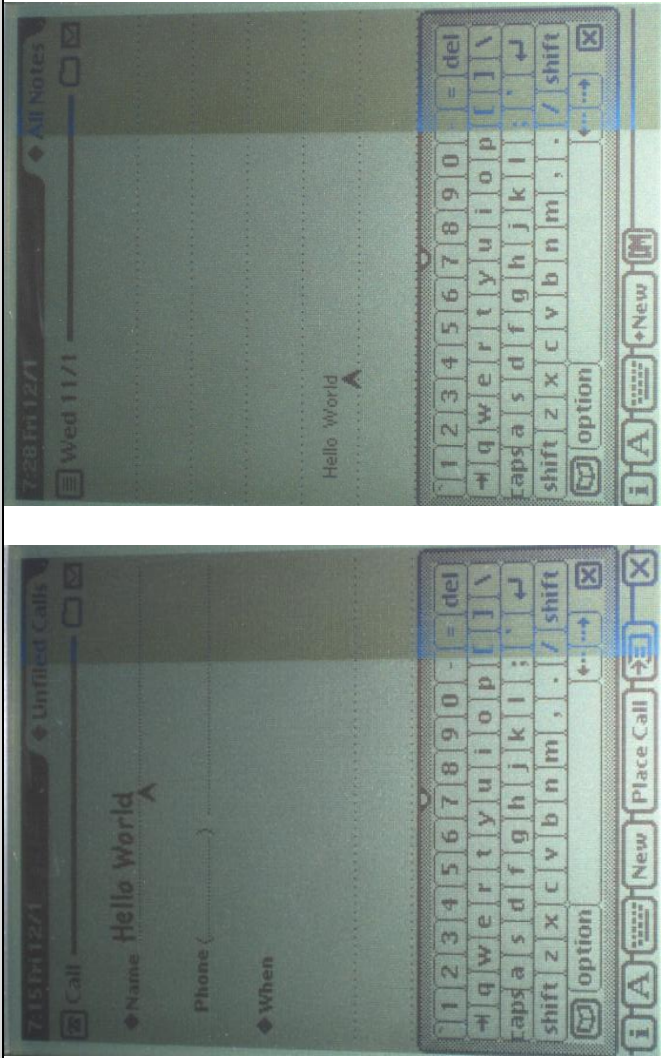
Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴</p> <p>In addition, the combination of the MessagePad 120 and U.S. Patent No. 5,148,155 (Martin ‘155) renders this claim obvious. Martin ‘155 teaches input methods which are independent of any application program. See Martin ‘155 Col. 16, ll. 30-32. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.2	in response to actuation of the actuatable icon, displaying the input method list;	<p>The screen cover of the MessagePad 120 describes the keyboard icon as follows:</p>  <p>As shown below, when the keyboard icon is pressed twice, the input method list appears:</p>

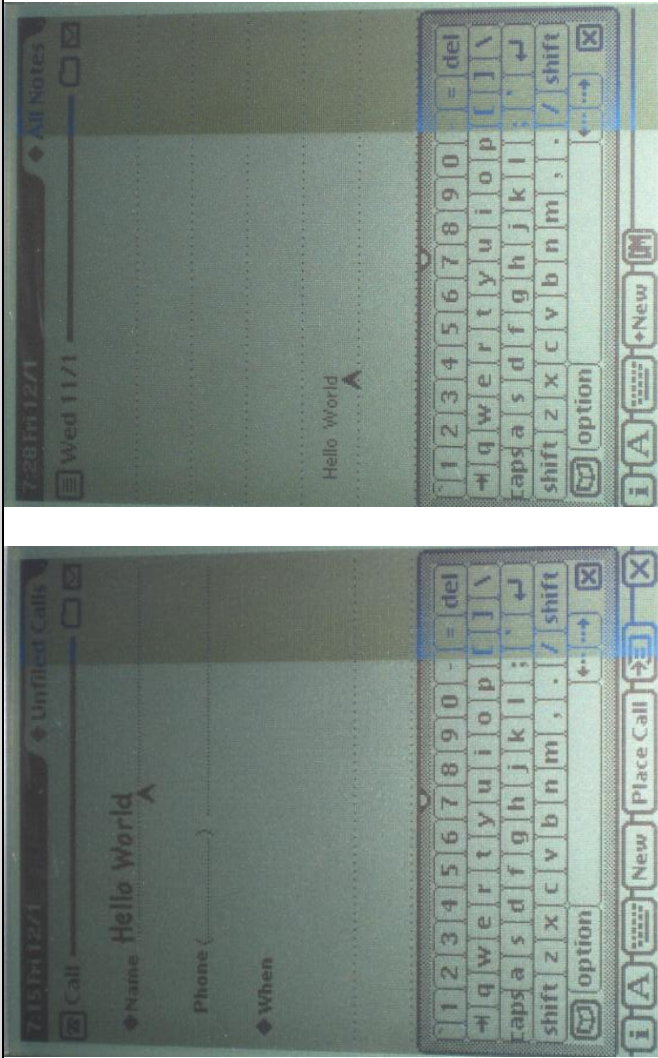
Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴</p>
1.3	receiving a selection of an input method from the input method list;	<p>The MessagePad 120 displays a keyboard icon along the status bar. As shown below, when pressed once, an on-screen keyboard appears.</p> 

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴</p> <p>When pressed twice, the other available input methods are shown and can be selected.</p> 
1.4	installing an input method component that corresponds to the selected input method, the input method component causing an interactive input panel to be displayed;	<p>The MessagePad 120 displays a keyboard icon along the status bar. As shown below, when pressed once, an on-screen keyboard is installed.</p>

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴</p>
		<div data-bbox="321 590 813 984">  </div> <p>As shown below, once the keyboard input method is installed, the input method list shows that the keyboard input method has been installed.</p> <div data-bbox="971 342 1463 1234">  </div>

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴</p>
		<p>The keyboard input method, once installed, sends keystrokes to the active application. As shown below, the phrase “Hello World” appears in the Notes and Calls application windows when typed from the on-screen keyboard.</p> <div data-bbox="446 804 1101 1312">  </div> <div data-bbox="446 262 1101 766">  </div>
1.5	receiving input via the interactive input panel; and	As shown below, the phrase “Hello World” appears in the Notes and Calls application windows when typed from the on-screen keyboard.

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide² and/or The NewtonScript Programming Language³ and/or Brockschmidt⁴</p> 
1.6	providing the input to a computer program of the one or more computer programs as if the information was received via user input received from a hardware input device.	<p>Using the soft keyboard of the MessagePad 120, a user can send keystrokes to the active application. As shown below, the phrase “Hello World” appears in the Notes and Calls application windows when typed from the on-screen keyboard. The phrase “Hello World” appears on the application window as if the user had typed this phrase using a traditional keyboard.</p>

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide ² and/or The NewtonScript Programming Language ³ and/or Brockschmidt ⁴
		

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
2	2. The method of claim 1 wherein providing the input to the computer program comprises communicating information representative of the input to	See Claims 1.5 & 1.6.

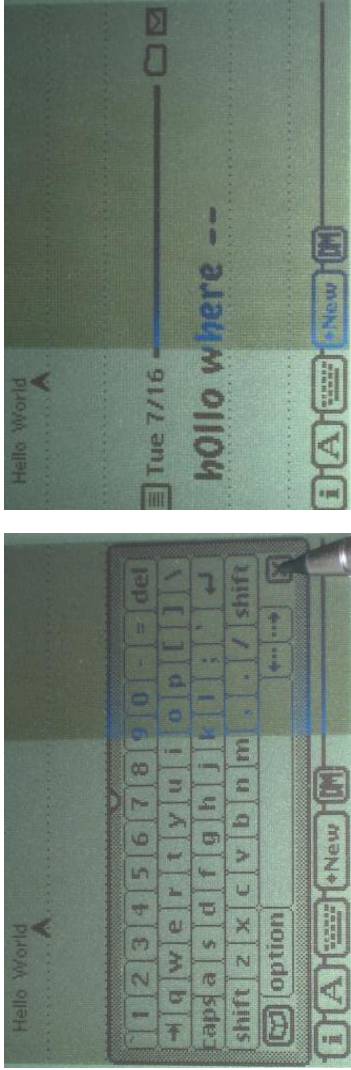
Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt
	a graphical windowing environment.	

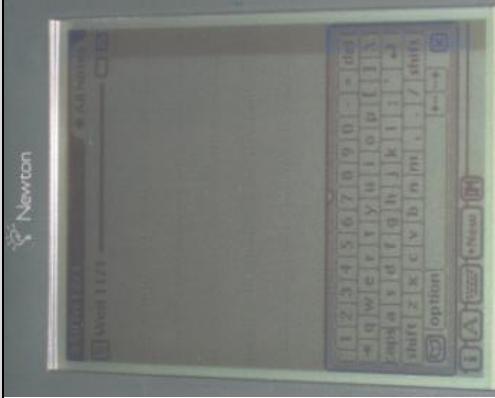
Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt
3	3. The method of claim 2 wherein communicating the information comprises passing the information to an interface.	<p>See Claim 1.5.</p> <p>Because the on-screen keyboard is separate from the Notepad and Calls applications, the input data must pass through an interface before reaching the application.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design an input interface system between an input method and an application to permit any input method to work with virtually any application.</p> <p>In addition, the combination of the MessagePad 120 and U.S. Patent No. 5,148,155 (“Martin ‘155”) renders this claim obvious. Martin ‘155 discloses a system with an interface between a plurality of input methods and the active application program. See Martin ‘155 Col 3, ll. 37-40. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt

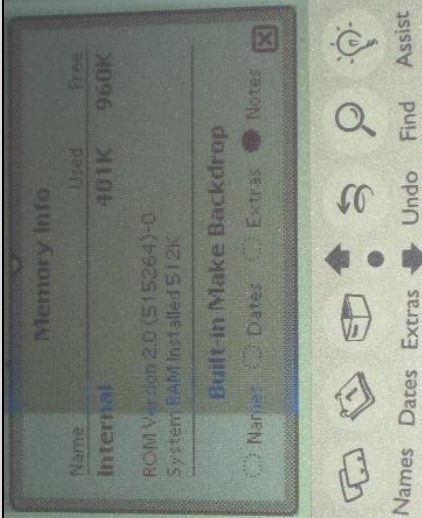

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
4	4. The method of claim 2 further comprising, communicating the information from the graphical windowing environment to an application program, wherein the computer program includes the application program, wherein the information is provided to the application program in a same manner as if the input was received via a hardware keyboard.	See Claims 1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
6	6. The method of claim 1 wherein the selected input method corresponds to a displayed keyboard, and wherein receiving input via the interactive input panel that corresponds to the selected input method comprises receiving	See Claims 1.5 & 1.6.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
	information corresponding to a keyboard character entered via the displayed keyboard.	
Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
8	8. The method of claim 1 further comprising, hiding the input panel.	When the close button of the input method window is depressed, the input method window closes. 
Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
9	9. The method of claim 1 further comprising, docking the input panel.	See Claim 1.1, which illustrates the docking of the input panel as follows:

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt</p>
		 <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel.</p>

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt</p>
10	10. At least one computer-readable medium having computer-executable instructions, which when executed perform the method of claim 1.	<p>The MessagePad 120 has a computer-readable medium.</p>


Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
		 

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
11	11. At least one computer-readable medium having computer-executable instructions stored thereon, which when executed by a computer system perform steps, comprising:	See Claims 1.1 & 10.
11.1	selecting one of a plurality of executable input methods for supplying user input to the computer system, wherein each executable input method is an interchangeable software component distinct from one or more application	See Claim 1.1.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
	programs, each executable input method having a defined interface set such that the executable input method is connectable to the application programs; opening an input window on a display of the computer system independent of a window of an active application program; and	
11.2		See Claim 1.1.
11.3	displaying an interactive input panel in the input window, the interactive input panel corresponding to the selected executable input method such that information corresponding to user input received by the selected executable input method via the interactive input panel is provided to the active application program as if the information was received via user input at a hardware input device.	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
13	13. The computer-readable medium of claim 11 further comprising, providing a	See Claim 1.1.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
	Software Input Panel (SIP) menu button on the display of the computer system, the SIP menu button being actuatable to display a selectable list of the plurality of executable input methods.	

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
14	14. The computer-readable medium of claim 13 further comprising, receiving a selection of one of the plurality of executable input methods displayed in the list as a selected executable input method, and in response, closing any open input window, and opening a new input window corresponding to the selected executable input method.	<p>As shown below, when the user selects the Numeric input method option, the Typewriter input method window closes, and the Numeric input method window opens.</p> 

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
15	15. At least one computer-readable medium having computer-executable instructions, which when executed perform steps, comprising:	See Claims 1.1 & 10.
15.1	presenting icons corresponding to a plurality of input methods available for a computer application, wherein each input method is a computer-executable software component distinct from the computer application;	See Claim 1.1
15.2	invoking a selected input method in response to a user selecting an icon corresponding to the selected input method, including presenting an input panel window; and	See Claim 1.1.
15.3	accepting user data entered in the input panel window for the computer application, wherein the user data is provided to the computer application as if the user data was received from a hardware input device.	See Claim 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
16	16. The computer-readable medium of claim 15 wherein accepting user data	See Claim 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt
	includes detecting user interaction with a touch-sensitive display.	

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt
17	17. The computer-readable medium of claim 15 wherein each input method comprises a component object model (COM) object, and wherein the step of invoking the selected input method includes the step of instantiating the COM object.	<p>NewtonScript, the programming language used to write software for the MessagePad 120, is built on an object model. See The NewtonScript Programming Language⁵, page 1-2:</p> <p>The Object Model</p> <p>NewtonScript is built on an object model. All data is stored as objects, or typed pieces of data. This differs from other object-oriented languages like C++ or Object Pascal, where data is a hybrid of objects and regular data types.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I</p>

⁵ Apple Press, Apple Computer Inc., The NewtonScript Programming Language, Addison Wesley Publishing Company (1996).

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt</p> <p>described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of the MessagePad 120 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
-------	-----------------------	--

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt
18	18. The computer-readable medium of claim 15 further comprising converting the user data to a Unicode character value.	<p>The Newton programming environment supports Unicode. See Newton Programmer’s Guide at 1-9:</p> <p style="padding-left: 40px;">The Newton text imaging facility supports Unicode directly, so the system can be easily localized to display languages using different script systems. The system is extensible, so it’s possible to add additional fonts, font engines, and printer drivers.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convert the user data to a Unicode character value.</p> <p>In addition, the combination of the MessagePad 120 and U.S. Patent No. 5,454,046 (“the Carmen Reference”)⁶ renders this claim obvious. See Carmen Reference, Col. 10, ll. 14-22 (“The text represented at block 62 may consist of one or more words represented in Unicode, a 16-bit character representation defined by the Unicode Consortium of Mountain View, Calif. This representation is used instead of ASCII or ANSI representations, since Unicode can accommodate over 65,000 character representations, representing the definition of character glyphs of virtually all languages of the world, whereas ASCII and ANSI representations are limited to 256 characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>See Figure 3A of Carmen Reference.</p>

⁶ “Universal Symbolic Handwriting Recognition System” issued to Carman; filed September 17, 1993; issued September 26, 1995.

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt</p>
		<div data-bbox="324 478 756 989"> </div> <p style="text-align: center;"><u>FIG. 3A</u></p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>Similarly, the combination of the MessagePad 120 and U.S. Patent No. 5,455,901 (“the Friend Reference”)⁷ renders this claim obvious. See Claim 6 of Friend Reference (“The device of claim 5 wherein said translator comprises a generator for generating machine readable characters from said strokes.”); Claim 7 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise Unicode characters.”); Claim 8 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise ASCII characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field</p>

⁷ “Input Device With Deferred Translation” issued to Friend; filed September 12, 1994; issued October 3, 1995.



Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
		and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
19	19. In a computing environment, a system comprising,	See Claim 1.
19.1	a manager component stored on one or more computer-readable media and configured: to manage selection of a selected input method from one or more available stored input methods, wherein each input method is a computer-executable software component distinct from one or more computer programs, and	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
	to send input data corresponding to a user input received at the selected input method to a graphical windowing environment; and	
19.2	the graphical windowing environment to receive the input data and to send the input data to a computer program of the one or more computer programs, wherein the input data is sent to the computer program as if the input data was received via user input received from a hardware input device.	See Claims 1.1 & 1.5.
Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
20	20. The system of claim 19 wherein the computer program comprises an application program having focus.	See Claims 1.1 & 1.5.
Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
21	21. The system of claim 19 further	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
	comprising an input panel window corresponding to the selected input method.	

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
22	22. The system of claim 21 wherein the selected input method presents an image representing a keyboard on the input panel window.	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
23	23. The system of claim 21 wherein the manager component selectively displays and hides the input panel window.	See Claim 14. See also page 24 of Newton Apple MessagePad Handbook, which provides: To use a different keyboard, tap the Keyboard button  a second time and from the list that appears, tap your choice. When you are finished using the on-screen keyboard, tap  on the bottom right to close it.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
25	25. The system of claim 19 where the input method is displayed on a touch-sensitive display screen.	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
26	26. The system of claim 19 wherein the manager component transfers information from the computer program to the selected input method.	See Claims 1.1 & 1.5. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design an input manager capable of transferring information from the computer program to the selected input method. In addition, the combination of MessagePad 120 and U.S. Patent No. 5,148,155 (Martin '155) renders this claim obvious. Martin '155 teaches the transferring information from the computer program to the selected input method. See Martin '155 Col. 32, ll. 36-44. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application),

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
		disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
27	27. The system of claim 19 wherein the selected input method calls functions in the manager component via a defined interface set.	See Claims 1.1 & 1.5. To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system capable of calling functions in the manager component via a defined interface set. In addition, the combination of MessagePad 120 and U.S. Patent 5,157,384 ("Greanias '384") renders this claim obvious. Greanias '384 teaches calling functions in the manager component via a defined interface set. See Greanias '384 Claim 37 and Col. 8, ll. 59-65. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified,

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt
		predictable solutions.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt
28	28. The system of claim 19 wherein the selected input method comprises an object.	<p>NewtonScript, the programming language used to write software for the MessagePad 120, is built on an object model. See The NewtonScript Programming Language, page 1-2:</p> <p>The Object Model</p> <p>NewtonScript is built on an object model. All data is stored as objects, or typed pieces of data. This differs from other object-oriented languages like C++ or Object Pascal, where data is a hybrid of objects and regular data types.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p>

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
		In addition, the combination of the MessagePad 120 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 ("The main reason why COM uses a different paradigm is that it's trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]"). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.
		Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
29	29. The system of claim 19 wherein the selected input method draws an input panel in an input panel window displayed in the graphical windowing environment.	See Claims 1.1 & 1.5.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
30	30. The system of claim 29 wherein the manager component selectively displays and hides the display of the input panel window.	See Claim 23.

Claim	U.S. Patent 7,411,582	Apple Newton MessagePad 120 ("MessagePad 120") alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer's Guide and/or The NewtonScript Programming Language and/or Brockschmidt
31	31. The system of claim 29 wherein the manager component docks the input panel window.	See Claim 1.1, which illustrates the docking of the input panel as follows:

Claim	U.S. Patent 7,411,582	<p>Apple Newton MessagePad 120 (“MessagePad 120”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or U.S. Patent 5,157,384 and/or Newton Programmer’s Guide and/or The NewtonScript Programming Language and/or Brockschmidt</p>
		<div data-bbox="321 531 813 926" data-label="Image"> </div> <p>See also page 55 of Newton Apple MessagePad Handbook, which provides:</p> <p>To move a keyboard, tap and hold down the pen on the picture hanger on the keyboard slip and drag to it where you want the keyboard on the screen.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel.</p>

EXHIBIT K

Exemplar Chart of U.S. Patent 7,411,582**U.S. Patent 5,252,951 (“Tannenbaum ‘951”)¹
Claims 1-4, 6, 8-11, 13-23, 25-31**

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
1	1. In a computing environment, a computer-implemented method comprising:	“An advanced user interface for use with a computer system operating on an integrated operating environment.” Abstract.
1.1	displaying an actuatable icon representative of an input method list that includes one or more selectable input methods for one or more computer programs, wherein each input method is a computer-executable software	“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract. “It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 38-40.

NOTE: Motorola applies the prior art in light of Microsoft’s improper assertions of infringement and improper application of the claims. Motorola does not agree with Microsoft’s application of the claims, or that the claims satisfy 35 U.S.C. § 112. Motorola’s contentions herein are not, and should in no way be seen as, admissions or adoptions as to any particular claim scope or construction, or as any admission that any particular element is met in any particular way. Motorola objects to any attempt to imply claim construction from the foregoing chart. Motorola’s prior art invalidity contentions are made in a variety of alternatives and do not represent Motorola’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim contained therein.

¹ “Graphical User Interface With Gesture Recognition In A Multiapplication Environment” issued to Tannenbaum et al.; filed October 21, 1991; issued October 12, 1993.

² Kraig Brockschmidt, Inside OLE (2d ed. 1995).

³ Upon information and belief, the MessagePad 120 was first sold in the United States in January of 1995. The Newton Apple MessagePad Handbook would have been distributed to the public around the same time.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
	component distinct from the computer programs;	<p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i>, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 3-12 (emphasis added).</p> <p>The input methods are each separately executable from the application programs. Col. 8, ll. 65-67 (“As shown, the AUI utility 109 can be written as a separate application program[.]”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement an actuatable icon representative of an input method list. For example, the invention discloses several types of interface utilities 109, and it would have been obvious to one of skill in the art to include this step to permit the user to select one of those input utilities. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p> <p>In addition, the combination of Tannenbaum ‘951 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the presentation of an actuatable icon representative of an input method list to a user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.2	in response to actuation of the actuatable icon, displaying the input method list;	See Claim 1.1 regarding the actuation of the actuatable icon.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
		<p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract.</p> <p>“It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 38-40.</p> <p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to present an input list to a user. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p> <p>In addition, the combination of Tannenbaum ‘951 and the MessagePad 120 renders this claim obvious. The MessagePad 120 teaches the presentation of an input list to a user. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>
1.3	receiving a selection of an input method from the input method list;	See Claim 1.2.
1.4	installing an input method component that corresponds to the selected input method, the input method component causing an interactive input panel to be displayed;	<p>A user can install an input method by using the appropriate method. Col. 9, ll. 9-12. (“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.”).</p> <p>Once installed, the input method can use the translation capabilities of the AUI 100 to communicate with the active application. Col. 8, ll. 3. See also Col. 9, ll. 12-19 (“A user can quickly enter columns of alphanumeric information directly on the screen by</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
		touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard. The space needed for the keyboard or keypad is quite small, and will not completely obscure the work area of the spreadsheet, word processor, etc., which is the active application program.”).
1.5	receiving input via the interactive input panel; and	Once installed, the input method can use the translation capabilities of the AUI 100 to communicate with the active application. Col. 8, ll. 3. See also Col. 9, ll. 12-19 (“A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard. The space needed for the keyboard or keypad is quite small, and will not completely obscure the work area of the spreadsheet, word processor, etc., which is the active application program.”).
1.6	providing the input to a computer program of the one or more computer programs as if the information was received via user input received from a hardware input device.	<p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code. The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface.” Abstract.</p> <p>“The alternative input subsystem module provides communication between the attached user friendly input devices and the remainder of the advanced user interface as well as application programs through the integrated operating environment. The alternative input subsystem module receives the input signals generated by the input devices and translates them to input messages useable in the advanced user interface.” Col. 3, ll. 65-68 to Col. 4, ll. 1-5.</p> <p>The AUI manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs. Col. 8, ll. 3.</p> <p>“After receiving an input message from Presentation Manager.TM. 156 together with information about which is the active application program, the PM-Link 201 refers back to the alternate input subsystem 203 to determine whether the “keyboard” or “mouse” message it received is in fact a keyboard or a mouse message, or rather a</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt ² and/or the MessagePad 120 ³
		touch, voice, image, or other message. The PM-Link 201 then passes the true message to the appropriate application program.” Col. 14, ll. 7-16.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
2	2. The method of claim 1 wherein providing the input to the computer program comprises communicating information representative of the input to a graphical windowing environment.	“As shown [in Figure 3], the AUI utility 109 can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.” Col. 8, ll. 65-68 to Col. 9, ll. 1-2.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
3	3. The method of claim 2 wherein communicating the information comprises passing the information to an interface.	The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs.” Col. 8, ll. 3. “The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.” Abstract.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		“The alternative input subsystem module provides communication between the attached user friendly input devices and the remainder of the advanced user interface as well as application programs through the integrated operating environment. The alternative input subsystem module receives the input signals generated by the input devices and translates them to input messages useable in the advanced user interface.” Col. 3, ll. 65-68 to Col. 4, ll. 1-5.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
4	4. The method of claim 2 further comprising, communicating the information from the graphical windowing environment to an application program, wherein the computer program includes the application program, wherein the information is provided to the application program in a same manner as if the input was received via a hardware keyboard.	<p>The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. See Figure 3. The AUI has “translation capabilities” for sending user data to the application programs.” Col. 8, ll. 3.</p> <p>“Another recent trend is to provide some sort of integration of computer program applications. Without integration, the user must employ separate application programs for word processing, database manipulation, graphics and electronic mail functions, and so forth. It is often quite difficult to integrate the outputs of the different programs into a single desired output. One solution has been to write a single integrated piece of software which incorporates a variety of applications which is called a multiple-function program.” Col. 2, ll. 43-52.</p> <p>“The interface profile module 104 is comprised of sets of application profiles 105 and the user profiles 107, which are files which list input messages produced by the AIS 103 from the input signals received by input devices 36, 38, 40, mapped to keyboard, mouse or other commands which are usable by existing application programs, e.g., mouse clicks, keystroke messages, MACROs, utility programs, etc.” Col. 8, ll. 55-62.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
6	6. The method of claim 1 wherein the selected input method corresponds to a displayed keyboard, and wherein receiving input via the interactive input panel that corresponds to the selected input method comprises receiving information corresponding to a keyboard character entered via the displayed keyboard.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.” Col. 9, ll. 9-16.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
8	8. The method of claim 1 further comprising, hiding the input panel.	See Claims 1.1 – 1.6. <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include . . . an image magnifying utility . . . The image magnifying utility will magnify a rectangle of fixed size around a point at which the appropriate gesture was made. The utility allows very accurate positioning of a cursor in the expanded image. After stylus liftoff, the normal size display is restored, and the selected cursor coordinates are sent to the active application program.” Col. 9, ll. 3-25.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to hide the input panel upon user command or upon the opening of another input window. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p> <p>In addition, the combination of Tannenbaum ‘951 and U.S. Patent No. 5,148,155 (“Martin ‘155”) renders this claim obvious. Martin ‘155 teaches the hiding of an input panel. See Martin ‘155 Abstract and Col. 17, ll. 11-22. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		<p>application program), address the same problem, disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
9	9. The method of claim 1 further comprising, docking the input panel.	<p>See Claim 23 of Tannenbaum ‘951 (“The computer system as recited in claim 22, which further comprises a touch input device disposed over the viewing surface of the display device, and wherein the advanced user interface further comprises <i>a pop-up keyboard module which displays an image of a keyboard on the display device [.]</i>”) (emphasis added).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p> <p>In addition, the combination of Tannenbaum ‘951 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified,</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
10	10. At least one computer-readable medium having computer-executable instructions, which when executed perform the method of claim 1.	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 6, ll. 40-51.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
11	11. At least one computer-readable medium having computer-executable instructions stored thereon, which when executed by a computer system perform steps, comprising:	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 6, ll. 40-51.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
11.1	selecting one of a plurality of executable input methods for supplying user input to the computer system, wherein each executable input method is an interchangeable software component distinct from one or more application programs, each executable input method having a defined interface set such that the executable input method is connectable to the application programs;	<p>See Claims 1.1 – 1.6.</p> <p>The invention provides for multiple input types which supply user input to the computer system. See Figure 3.</p> <p>The input methods are each separately executable from the application programs. Col. 8, ll. 65-67 (“As shown, the AUI utility 109 can be written as a separate application program[.]”).</p> <p>The input methods are connectable to the application programs through an advanced user interface and/or the operating system. Col. 1, ll. 15-19 (this invention “relates to an advanced user interface which allows a user to select one or more input devices to input data into a computer running a program originally written for a different input device in a multiapplication environment”). See also Figure 3.</p> <p>“The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.” Abstract.</p>
11.2	opening an input window on a display of the computer system independent of a window of an active application program; and	<p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.</p> <p>The AUI Utility is independent of the application program. Col. 8, ll. 8-13 Col. 8, ll. 65-68 to Col. 9, ll. 1-2 (“the AUI utility 109 can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.”).</p>
11.3	displaying an interactive input panel in the input window, the interactive input panel corresponding to the selected executable input method such that information corresponding to user input	<p>See Claims 1.1 – 1.6.</p> <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
	received by the selected executable input method via the interactive input panel is provided to the active application program as if the information was received via user input at a hardware input device.	<p>“pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i>, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 3-12 (emphasis added).</p> <p>The input methods are each separately executable from the application programs. Col. 8, ll. 65-67 (“As shown, the AUI utility 109 can be written as a separate application program[.]”).</p> <p>“After receiving an input message from Presentation Manager.TM. 156 together with information about which is the active application program, the PM-Link 201 refers back to the alternate input subsystem 203 to determine whether the “keyboard” or “mouse” message it received is in fact a keyboard or a mouse message, or rather a touch, voice, image, or other message. The PM-Link 201 then passes the true message to the appropriate application program.” Col. 14, ll. 7-16.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
13	13. The computer-readable medium of claim 11 further comprising, providing a Software Input Panel (SIP) menu button on the display of the computer system, the SIP menu button being actuatable to display a selectable list of the plurality of executable input methods.	<p>“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract.</p> <p>“It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 38-40.</p> <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		gesture or touching a keyboard <i>icon</i> , the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 3-12 (emphasis added).
		The input methods are each separately executable from the application programs. Col. 8, ll. 65-67 (“As shown, the AUI utility 109 can be written as a separate application program[.]”).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
14	14. The computer-readable medium of claim 13 further comprising, receiving a selection of one of the plurality of executable input methods displayed in the list as a selected executable input method, and in response, closing any open input window, and opening a new input window corresponding to the selected executable input method.	See Claim 8. “A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i> , the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 3-12 (emphasis added).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
15	15. At least one computer-readable medium having computer-executable instructions, which when executed	“The preferred embodiment of the invention comprises a set of computer programs for controlling the interaction between a user and a computer system as shown in FIG. 2. The invention is primarily envisioned for use with a personal computer such as the

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
	perform steps, comprising:	IBM PS/2.TM.; however, the principles of this invention can be extended to other individual workstations or to much larger data processing systems. The architectural block diagram of FIG. 2 includes a central processing unit (CPU) 20 connected by means of a system bus 22 to a read-only memory (ROM) 24 and a random access memory (RAM) 26.” Col. 6, ll. 40-51.
15.1	presenting icons corresponding to a plurality of input methods available for a computer application, wherein each input method is a computer-executable software component distinct from the computer application;	<p>See Claims 1.1 – 1.6.</p> <p>“A few examples of utilities which might be found in the advanced user interface utilities 109 include a “pop-up” keyboard, a “pop-up” numeric keypad, an image magnifying utility and an electronic mail routing slip. The “pop-up” keyboard and “pop-up” keypad are functions which are preferably designed to operate with a touch input device at the display surface. By issuing the proper command, e.g., a circle gesture or touching a keyboard <i>icon</i>, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 3-12 (emphasis added).</p> <p>The AUI Utility is independent of the application program. Col. 8, ll. 8-13 Col. 8, ll. 65-68 to Col. 9, ll. 1-2 (“the AUI utility 109 can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.”).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to implement an actuable icon representative of an input method list. For example, the invention discloses several types of interface utilities 109, and it would have been obvious to one of skill in the art to include this step to permit the user to select one of those input utilities. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p>
15.2	invoking a selected input method in response to a user selecting an icon corresponding to the selected input method, including presenting an input panel window; and	<p>See Claims 1.1 – 1.6.</p> <p>“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.”</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		Col. 9, ll. 9-16.
15.3	accepting user data entered in the input panel window for the computer application, wherein the user data is provided to the computer application as if the user data was received from a hardware input device.	See Claims 15.2.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
16	16. The computer-readable medium of claim 15 wherein accepting user data includes detecting user interaction with a touch-sensitive display.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen. A user can quickly enter columns of alphanumeric information directly on the screen by touching the keys displayed on the screen, thus eliminating the need to divert their attention to manual keyboard.” Col. 9, ll. 9-16. See Claim 16 of Tannenbaum ‘951 (“The computer system as recited in claim 13, which further comprises a gesture recognition unit which interprets the input signals transmitted by a touch input device in response to a user drawing symbols on a surface detected by the touch input device.”).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
17	17. The computer-readable medium of	To the extent not expressly or inherently disclosed, it would have been obvious to one

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
	claim 15 wherein each input method comprises a component object model (COM) object, and wherein the step of invoking the selected input method includes the step of instantiating the COM object.	<p>of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Tannenbaum ‘951 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
18	18. The computer-readable medium of claim 15 further comprising converting the user data to a Unicode character value.	<p>“The advanced user interface includes alternate input modules which translate the input signals transmitted from the various input devices into input messages useable by the rest of the interface. The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.” Abstract.</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to convert the user data to a Unicode character value. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p> <p>In addition, the combination of Tannenbaum ‘951 and U.S. Patent No. 5,454,046 (“the Carmen Reference”)⁴ renders this claim obvious. See Carmen Reference, Col. 10, ll. 14-22 (“The text represented at block 62 may consist of one or more words represented in Unicode, a 16-bit character representation defined by the Unicode Consortium of Mountain View, Calif. This representation is used instead of ASCII or ANSI representations, since Unicode can accommodate over 65,000 character representations, representing the definition of character glyphs of virtually all languages of the world, whereas ASCII and ANSI representations are limited to 256 characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>See Figure 3A of Carmen Reference.</p>

⁴ “Universal Symbolic Handwriting Recognition System” issued to Carman; filed September 17, 1993; issued September 26, 1995.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		<div data-bbox="289 478 717 987"> </div> <p style="text-align: center;">FIG. 3A</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p> <p>Similarly, the combination of Tannenbaum ‘951 and U.S. Patent No. 5,455,901 (“the Friend Reference”)⁵ renders this claim obvious. See Claim 6 of Friend Reference (“The device of claim 5 wherein said translator comprises a generator for generating machine readable characters from said strokes.”); Claim 7 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise Unicode characters.”); Claim 8 of Friend Reference (“The device of claim 6 wherein said machine readable characters comprise ASCII characters.”). One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (user input systems which are independent of an</p>

⁵ “Input Device With Deferred Translation” issued to Friend; filed September 12, 1994; issued October 3, 1995.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		application program), address the same problem (permitting an input method to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
19	19. In a computing environment, a system comprising,	“An advanced user interface for use with a computer system operating on an integrated operating environment.” Abstract.
19.1	a manager component stored on one or more computer-readable media and configured: to manage selection of a selected input method from one or more available stored input methods, wherein each input method is a computer-executable software component distinct from one or more computer programs, and to send input data corresponding to a user input received at the selected input method to a graphical windowing	“The advanced user interface allows a user to select among user-friendly input devices to operate any application program according to his individual preferences without change to the application program code.” Abstract. “It is therefore an object of the invention to allow a user to select between a plurality of input devices to input data into a computer system.” Col. 3, ll. 38-40. The AUI serves as the manager component. It manages the selection of the input method and communicates the input data of the input method to the application window. <i>See</i> Fig. 3. The input methods are each separately executable from the application programs. Col. 8, ll. 65-68 to Col. 9, ll. 1-2 (“the AUI utility 109 can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.”).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
	environment; and	The AUI has “translation capabilities” for sending user data to the application programs. Col. 8, ll. 3.
19.2	the graphical windowing environment to receive the input data and to send the input data to a computer program of the one or more computer programs, wherein the input data is sent to the computer program as if the input data was received via user input received from a hardware input device.	<p>The Integrated Operating Environment (windowing environment) serves as the interface between the AUI and the applications. See Figures 3 & 6; see also Abstract (“The advanced user interface also includes interface profiles which contain mappings of the input messages against corresponding commands useable by the application programs, the integrated operating environment or other modules of the advanced user interface itself.”).</p> <p>The AUI Utility is independent of the application program. Col. 8, ll. 8-13 Col. 8, ll. 65-68 to Col. 9, ll. 1-2 (“the AUI utility 109 can be written as a separate application program, or set of application programs, so that it can use the message passing capabilities of the integrated operating environment 56 to communicate with other application programs.”).</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
20	20. The system of claim 19 wherein the computer program comprises an application program having focus.	“The integrated operating environment allows a plurality of application programs to be running simultaneously, one of which is designated the active application program to which all input data is directed.” Abstract.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
21	21. The system of claim 19 further	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon,

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
	comprising an input panel window corresponding to the selected input method.	the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
22	22. The system of claim 21 wherein the selected input method presents an image representing a keyboard on the input panel window.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
23	23. The system of claim 21 wherein the manager component selectively displays and hides the input panel window.	See Claim 8. “By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
25	25. The system of claim 19 where the	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon,

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
	input method is displayed on a touch-sensitive display screen.	the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
26	26. The system of claim 19 wherein the manager component transfers information from the computer program to the selected input method.	See Figure 8. “After being recognized by the gesture recognition unit 213 as a circle gesture, the PM-Link 201 passes the circle gesture to the spreadsheet 150. The spreadsheet returns an “R0” message at 285, indicating that the circle gesture was not understood. Since the circle gesture was not understood at 287, the PM-Link 201 refers to the application profile 205 for the spreadsheet 150 at 289 and finds no corresponding command for the circle gesture. The PM-Link 201 then refers to the user profile 207 at 291 which contains the corresponding command “invoke the pop-up keyboard”. As the command from the user profile 207 has the highest priority at 293, the PM-Link 201 sends this message to the AUI utilities module 209 at 295 and the pop-up keyboard is presented to the user on the display.” Col. 17, ll. 8-22.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
27	27. The system of claim 19 wherein the selected input method calls functions in the manager component via a defined interface set.	See Figure 4. See Claim 10 of Tannenbaum ‘951 (“The advanced user interface as recited in claim 9 wherein the interface profiles contain mappings of touch input messages against corresponding mouse commands.”).

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		“After determining the active application program, spreadsheet 50 at 119, the environment link 101 refers to the application profile 105 of the spreadsheet 50 for the command which corresponds to the input message “GRAPH” which will be recognized by the spreadsheet 50, i.e., the menu selection-- graph--.” Col. 9, ll. 48-54.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
28	28. The system of claim 19 wherein the selected input method comprises an object.	<p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to design a system whereby the input methods are objects or COM objects. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2. Well before the priority date of the ‘582 patent, those skilled in the art were aware of the need to develop component software as a means to upgrade applications in lieu of updating the source code of those applications; they were also aware of OLE as an available method.. See Brockschmidt at page 1160 (“In Chapter 1 . . . I described the benefits [of using OLE] merely as the capability to add features to applications by purchasing components rather than by purchasing entirely new builds of feature-laden monolithic applications. In other words, instead of necessarily purchasing applications, end users would build their own custom applications to solve their own specific problems.”).</p> <p>In addition, the combination of Tannenbaum ‘951 and Brockschmidt renders this claim obvious. Brockschmidt teaches the use of COM objects to integrate various independent components with currently-existing applications. See Brockschmidt at page 253 (“The main reason why COM uses a different paradigm is that it’s trying to solve the problem of integrating components that are developed, deployed, and revised independently so that new components can be instantly and robustly integrated into a running system. COM solves problems at the runtime binary level, through the binary standard of interfaces, and is concerned with components and objects that live and execute outside the boundaries of an application[.]”). One of skill in the art would be</p>

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
		motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period. Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
29	29. The system of claim 19 wherein the selected input method draws an input panel in an input panel window displayed in the graphical windowing environment.	“By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
30	30. The system of claim 29 wherein the manager component selectively displays and hides the display of the input panel window.	See Claim 8. “By issuing the proper command, e.g., a circle gesture or touching a keyboard icon, the keyboard or keypad will “pop-up” on the screen.” Col. 9, ll. 9-12.

Claim	U.S. Patent 7,411,582	U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120
31	31. The system of claim 29 wherein the manager component docks the input panel window.	<p>U.S. Patent 5,252,951 (“Tannenbaum ‘951”) alone and/or in combination with U.S. Patent No. 5,454,046 and/or U.S. Patent No. 5,455,901 and/or U.S. Patent No. 5,148,155 and/or Brockschmidt and/or the MessagePad 120</p> <p>See Claim 9.</p> <p>See Claim 23 of Tannenbaum ‘951 (“The computer system as recited in claim 22, which further comprises a touch input device disposed over the viewing surface of the display device, and wherein the advanced user interface further comprises a pop-up keyboard module which displays an image of a keyboard on the display device [.]”) (emphasis added).</p> <p>To the extent not expressly or inherently disclosed, it would have been obvious to one of skill in the art to dock the input panel. Col. 3, ll. 21-35; Col. 24, ll. 53-68 to Col. 25, ll. 1-2.</p> <p>In addition, the combination of Tannenbaum ‘951 and the MessagePad 120 renders this claim obvious. The MessagePad 120 docks its soft keyboard. One of skill in the art would be motivated to combine these references because they are all in the same field and share common subject matter (component programs which are independent of an application program), address the same problem (permitting a component program to work with virtually any application), disclose the same or similar techniques, and were developed during the same general time period.</p> <p>Additionally, a person of ordinary skill in the art would have been motivated to combine the methods disclosed in these references because there was a design need and/or market pressure to do so, and there were a finite number of identified, predictable solutions.</p>

EXHIBIT 4

**UNITED STATES INTERNATIONAL TRADE COMMISSION
WASHINGTON, D.C.**

In the Matter of

**CERTAIN HANDHELD ELECTRONIC
COMPUTING DEVICES,
RELATED SOFTWARE, AND
COMPONENTS THEREOF**

Investigation No. 337-TA-769

**BARNES & NOBLE, INC.'S AND BARNESANDNOBLE.COM LLC'S
NOTICE OF PRIOR ART**

Pursuant to the Procedural Schedule (Order No. 4), ¶ 5 of the Ground Rules (Order No. 2), and 35 U.S.C. § 282, respondents Barnes & Noble, Inc. and Barnesandnoble.com LLC (collectively, Barnes & Noble) hereby provide the following preliminary Notice of Prior Art with respect to the claims of United States Patent Nos. 5,778,372 to Cordell et al. (the '372 patent), 5,889,522 to Chew et al. (the '522 patent), 6,339,780 to Shell et al. (the '780 patent), 6,891,551 to Keely et al. (the '551 patent), and 6,957,233 to Beezer et al. (the '233 patent) (collectively, the patents-in-suit) asserted by Microsoft Corp. ("Microsoft").

Barnes & Noble's discovery and investigations in connection with this action and the patents-in-suit are continuing, and thus this notice of prior art is preliminary in nature and based on information obtained to date. Document production is still ongoing, several important third party discovery requests directed to prior art are still outstanding, no depositions have occurred to date, and Barnes & Noble has raised concerns with Microsoft's interrogatory responses that affect prior art assertions in this case. Accordingly, Barnes & Noble reserves the right to further modify, amend, or supplement this identification of prior art in accordance with the Scheduling Order, the Ground Rules applicable in this case, and/or any other governing rules as this action progresses and additional information is obtained. Moreover, while the prior art references listed

below are categorized by patent-in-suit, the references listed in connection with one patent-in-suit also have bearing on the invalidity of the other patents-in-suit because they relate, among other things, to the general knowledge possessed by one of ordinary skill in the art and can be combined with the prior art listed in connection with the other patents to render their asserted claims obvious.

I. '372 Patent

A. Prior Art

- 11/24/1995 post to comp.infosystems.www.browsers.ms-windows entitled Netscape 2.0b3 – 1st Impressions; available at http://groups.google.com/group/comp.infosystems.www.browsers.ms-windows/msg/8602c21aced97841?dmode=source&utoken=u1cWlUgAAADdRJqY_P0SuDfxNi4Z9xuSleg-tYXS2MURtSjm1DwFncU02o5MxfE5BRk_lVaLL-DichQ7pGlIEJBR8dFHmiSnXccd9b8p7N6Z-iiZ0oFTBw
- 11/24/1995 post to comp.os.ms-windows.win95.misc entitled Netscape Navigator 2.0 Beta 3 is available; available at http://groups.google.com/group/comp.os.ms-windows.win95.misc/msg/46dd346380e35cb3?dmode=source&utoken=riuG1T0AAACn6AG3aRlQNI89iyczJ9SXM1wLGRatfk_VGOenGXhqFRCKZYL7ve7WV4r_e2Zn1S-3uFh9VErLNglO2zHbFEK4
- 11/26/1995 post to comp.infosystems.www.browsers.ms-windows entitled Some Notes from Netscape: Beta 3 Availability; http://groups.google.com/group/comp.infosystems.www.browsers.ms-windows/msg/9898c308c4a63e7e?dmode=source&utoken=BKLg6kgAAADdRJqY_P0SuDfxNi4Z9xuSleg-tYXS2MURtSjm1DwFnfYz8imBGi145Fwt5NvNMiJG3ebz8WNAOz3oD8X93TP_N7hzTudyT04UxYvjBIzM6Q
- 12/12/1995 post to comp.infosystems.www.misc entitled Re: netscape 1.1 vs 2.0b – backgrounds; available at http://groups.google.com/group/comp.infosystems.www.misc/msg/e9387e3e6fe1f771?dmode=source&utoken=I5tSQTkAAADdRJqY_P0SuDfxNi4Z9xuSgN6JXLI-vEteZzj_rGfjZ6cEz086CYigP1cO9gtpdtDMi70Y-X-_WbdDsWgsFCgj.
- 1996 'Excellence in Software' Award Winners Named, Netscape Corp., 3/1996, <http://home.netscape.com/newsref/pr/newsrelease101.html>

- Adobe and Netscape to Bring Commercial Publishing to the Internet, Netscape Corp., 3/1995, <http://home.netscape.com/newsref/pr/newsrelease20.1.html>
- Arena 0.96, W3C, <http://www.w3.org/Arena/0.96.html>
- Arena 0.97, W3C, <http://www.w3.org/Arena/0.97.html>
- Arena 0.98, W3C, <http://www.w3.org/Arena/0.98.html>
- Barnett, Guide to Netscape Navigator 2.0, Ziff-Davis Press, 1996, pp. i-292
- Berghetl, The client's side of the World-Wide Web, *Communications of the ACM*, 1/1996, Vol. 39, Issue 1, pp. 30-40.
- Carts, How to put the World Wide Web to work for you, *Laser Focus World*, 1/1995, pp. 116-117
- Catledge et al., Characterizing Browsing Strategies in the World-Wide Web, *Proceedings of the Third International World-Wide Web Conference*, 1995, Vol. 27, Issue 6, pp. 1-9
- Cheek, Web Tools: Joint Those Marking the Internet Work for Their Agency's Mission, *Government Computer News*, Vol. 15, N. 5, 3/1996, pp. 1-38
- Chen, Romancing Internet and Multimedia: Use of World Wide Web and Web Browsers, *Microcomputers for Information Management, An International Journal for Library Information Services*, Vol. 11, No. 1, 1994, pp. 307-313
- Controlling Document Backgrounds, Netscape Corp., http://home.netscape.com/assist/net_sites/bg/index.html (with linked materials)
- Cook, Write a Simple HTTP-based Server Using MFC and Windows Sockets, *Microsoft Systems Journal*, 2/1996
- EInet's MacWeb browser, including at least versions 0.98 alpha, 1.00 alpha 2, 1.00 alpha 2.1, 1.00 alpha 2.2, 1.00 alpha 3, 1.00 alpha 3.1, and 1.00 alpha 3.2
- Fekete, A multi-layer graphic model for building interactive graphical applications, *Proceedings of the Conference on Graphics Interface*, 1992
- Fifteen Companies Deliver First Plug-Ins for Netscape Navigator API, Netscape Corp., 1/1996, <http://home.netscape.com/newsref/pr/newsrelease73.html>
- Gerber, Browsers on the Wild, Wild Web, *Network Computing*, 4/1995, p. 54
- Greengard, Home, Home on the Web, *Personnel Journal*, Vol. 75, No. 3, 3/1996, pp. 1-9
- Hardin, MacWeb 1.x available at <http://www.ccadams.org/se/macweb2.html>

- Herbst, Webfest IV, A Report of the Happenings at the Fourth World-Wide Web Conference, *Internet World*, 3/1996, pp. 22-26
- Holden, Mastering Netscape 2.0 for Macintosh the Complete Guide to Netscape 2.0, Hayden Books, 1995, pp. i-619
- HTML tags sorted by Navigator Releases, Netscape Corp., http://developer.netscape.com/library/documentation/htmlguid/tags_complete.html
- HyperText Markup Language Specification Version 3.0, 3/1995, available at <http://www.w3.org/MarkUp/html3/html3.txt>
- James, Official Netscape Navigator 2.0 Book, Windows Edition: The Definitive Guide to the World's Most Popular Internet Navigator, Netscape Press, 1st Ed., 1995, pp. i-657
- JP 6036070 to Urabe et al. (2/1994)
- Kaashoek et al., Dynamic Documents: Mobile Wireless Access to the WWW, *Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications*, 12/1994, pp. 1-6.
- Kraynak, The Complete Idiot's Guide to Mosaic, Alpha Books, 1st Ed., 1/1995, pp. 193-205
- Lemay, Teach Yourself Web Publishing with HTML 3.0 in a Week, Sams Publishing, 12/1995
- Lichty et al., America Online's Internet, Macintosh Edition, Ventana Press, Inc., 1994, pp. i-282
- MacWeb Developer Notes, EINet, 3/1995
- MacWeb Features, EINet, 3/1995
- Milheim, Authoring-Systems Software for Computer-Based Training, *Education Tech. Pubs.*, 1994, pp. 77-79
- Morris, HTML for Fun and Profit, SunSoft Press, 1995, pp. 63-72
- NCSA Mosaic, including at least version 2.0.0 beta 4 (available 4/1995) and all versions available prior to 4/1996
- Netscape and Sun, Netscape Corp., 4/1995, <http://home.netscape.com/newsref/pr/newsrelease23.html>
- Netscape Announces 'Netscape Bugs Bounty' With Release of Netscape Navigator 2.0 Beta, Netscape Corp., 10/1995, <http://home.netscape.com/newsref/pr/newsrelease48.html>

- Netscape Announces Add-on Product Suite for Popular Netscape Navigator Software, Netscape Corp., 10/1995, <http://home.netscape.com/newsref/pr/newsrelease57.html>
- Netscape Announces Netscape Navigator Gold 2.0, Netscape Corp., 9/1995, <http://home.netscape.com/newsref/pr/newsrelease42.html>
- Netscape Announces Public Beta of Netscape Navigator Gold, Netscape Corp., 1/1996, <http://home.netscape.com/newsref/pr/newsrelease78.html>
- Netscape Communications Now Builds on Tradition of Freeware for the Net, http://news.cnet.com/Netscapes-original-browser-press-release/2030-1032_3-5406484.html, 10/1994
- Netscape Communications Offers New Network Navigator Free on the Internet, 10/1994, <http://web.archive.org/web/19970614003509/http://home.netscape.com/newsref/pr/newsrelease1.html>
- Netscape Communications Ships Release 1.0 of Netscape Navigator and Netscape Servers, 12/1994, <http://home.netscape.com/newsref/pr/newsrelease8.html>
- Netscape Handbook, Netscape Corp., 1995, <http://home.netscape.com/eng/mozilla/1.1/handbook/> (with linked materials)
- Netscape Handbook, Netscape Corp., 1995, <http://home.netscape.com/eng/mozilla/1.2/handbook/> (with linked materials)
- Netscape Introduces International Beta Versions of Netscape Navigator 2.0, Netscape Corp., 2/1996, <http://home.netscape.com/newsref/pr/newsrelease90.html>
- Netscape Introduces Netscape Navigator 2.0, Netscape Corp., 9/1995, <http://home.netscape.com/newsref/pr/newsrelease43.html>
- Netscape Navigator 1.1 Alpha 1 for Macintosh, Netscape Corp., <http://home.netscape.com/assist/relnotes/mac-1.1a1.html> (with linked materials)
- Netscape Navigator 1.1 alpha 1 for Unix, Netscape Corp., <http://home.netscape.com/assist/relnotes/unix-1.1a1.html> (with linked materials)
- Netscape Navigator 1.1 Alpha 1 for Windows, Netscape Corp., <http://home.netscape.com/assist/relnotes/windows-1.1a1.html> (with linked materials)
- Netscape Navigator 1.1 Beta 1 for Macintosh, Netscape Corp., <http://home.netscape.com/assist/relnotes/mac-1.1b1.html> (with linked materials)
- Netscape Navigator 1.1 beta 1 for Unix, Netscape Corp., <http://home.netscape.com/assist/relnotes/unix-1.1b1.html> (with linked materials)

- Netscape Navigator 1.1 Beta 1 for Windows, Netscape Corp., 4/1995,
<http://home.netscape.com/assist/relnotes/windows-1.1b1.html> (with linked materials)
- Netscape Navigator 1.1 Beta 2 for Macintosh, Netscape Corp.,
<http://home.netscape.com/assist/relnotes/mac-1.1b2.html> (with linked materials)
- Netscape Navigator 1.1 beta 2 for Unix, Netscape Corp.,
<http://home.netscape.com/assist/relnotes/unix-1.1b2.html> (with linked materials)
- Netscape Navigator 1.1 Beta 2 for Windows, Netscape Corp., 6/1995,
<http://home.netscape.com/assist/relnotes/windows-1.1b2.html> (with linked materials)
- Netscape Navigator 1.1 Beta 3 for Macintosh, Netscape Corp.,
<http://home.netscape.com/assist/relnotes/mac-1.1b3.html> (with linked materials)
- Netscape Navigator 1.1 beta 3 for Unix, Netscape Corp.,
<http://home.netscape.com/assist/relnotes/unix-1.1b3.html> (with linked materials)
- Netscape Navigator 1.1 Beta 3 for Windows, Netscape Corp., 6/1995,
<http://home.netscape.com/assist/relnotes/windows-1.1b3.html> (with linked materials)
- Netscape Navigator 1.1 for Macintosh, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.1/relnotes/mac-1.1.html> (with linked materials)
- Netscape Navigator 1.1 for Unix, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.1/relnotes/unix-1.1.html> (with linked materials)
- Netscape Navigator 1.1 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.1/relnotes/windows-1.1.html> (with linked materials)
- Netscape Navigator 1.1II for Unix, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.1/relnotes/unix-1.1I.html>
- Netscape Navigator 1.12 for Macintosh, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.1/relnotes/mac-1.12.html> (with linked materials)
- Netscape Navigator 1.12 for Unix, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.1/relnotes/unix-1.12.html> (with linked materials)
- Netscape Navigator 1.12I for Unix, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.1/relnotes/unix-1.12I.html>
- Netscape Navigator 1.1a1, 1.1b1, 1.1b2, 1.1b3, 1.1, 1.1N, 1.1II, 1.12, 1.12I, 2.0a2, 2.0b1, 2.0b1J, 2.0b1N, 2.0b2, 2.0b3, 2.0b4, 2.0b5, 2.0b6a, and 2.0 for Unix, along with all other versions available prior to 4/1996.

- Netscape Navigator 1.1a1, 1.1b1, 1.1b2, 1.1b3, 1.1, 1.1N, 1.12, 2.0a2, 2.0b1, 2.0b2, 2.0b3, 2.0b4, 2.0b5, 2.0b6a, 2.0Jb1 and 2.0 for Macintosh, along with all other versions available prior to 4/1996.
- Netscape Navigator 1.1a1, 1.1b1, 1.1b2, 1.1b3, 1.1, 1.1N, 1.2b1, 1.2b2, 1.2b3, 1.2b4, 1.2b5, 1.2b6, 1.2, 1.22, 2.0a2, 2.0b1, 2.0b2, 2.0b3, 2.0b4, 2.0b5, 2.0b6a, and 2.0 for Windows, along with all other versions available prior to 4/1996.
- Netscape Navigator 1.1N for Unix, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.1/relnotes/unix-1.1N.html> (with linked materials)
- Netscape Navigator 1.2 beta 1 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.2/relnotes/windows-1.2b1.html>
- Netscape Navigator 1.2 beta 2 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.2/relnotes/windows-1.2b2.html>
- Netscape Navigator 1.2 beta 3 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.2/relnotes/windows-1.2b3.html>
- Netscape Navigator 1.2 beta 4 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.2/relnotes/windows-1.2b4.html>
- Netscape Navigator 1.2 beta 5 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.2/relnotes/windows-1.2b5.html>
- Netscape Navigator 1.2 beta 6 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.2/relnotes/windows-1.2b6.html>
- Netscape Navigator 1.2 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.2/relnotes/windows-1.2.html>
- Netscape Navigator 1.22 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/1.2/relnotes/windows-1.22.html>
- Netscape Navigator 2.0 for Macintosh, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/mac-2.0.html>
- Netscape Navigator 2.0 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/windows-2.0.html>
- Netscape Navigator 2.0 Java B1 for Macintosh, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0Java/relnotes/mac-2.0JavaB1.html>
- Netscape Navigator 2.0a2 for Macintosh, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/mac-2.0a2.html>

- Netscape Navigator 2.0a2 for Unix, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/unix-2.0a2.html>
- Netscape Navigator 2.0a2 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/windows-2.0a2.html>
- Netscape Navigator 2.0b1 for Macintosh, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/mac-2.0b1.html>
- Netscape Navigator 2.0b1 for Unix, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/unix-2.0b1.html>
- Netscape Navigator 2.0b1 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/windows-2.0b1.html>
- Netscape Navigator 2.0b2 for Macintosh, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/mac-2.0b2.html>
- Netscape Navigator 2.0b2 for Unix, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/unix-2.0b2.html>
- Netscape Navigator 2.0b2 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/windows-2.0b2.html>
- Netscape Navigator 2.0b3 for Macintosh, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/mac-2.0b3.html>
- Netscape Navigator 2.0b3 for Unix, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/unix-2.0b3.html>
- Netscape Navigator 2.0b3 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/windows-2.0b3.html>
- Netscape Navigator 2.0b4 for Macintosh, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/mac-2.0b4.html>
- Netscape Navigator 2.0b4 for Unix, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/unix-2.0b4.html>
- Netscape Navigator 2.0b4 for Windows, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/windows-2.0b4.html>
- Netscape Navigator 2.0b5 for Macintosh, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/mac-2.0b5.html>
- Netscape Navigator 2.0b5 for Unix, Netscape Corp.,
<http://home.netscape.com/eng/mozilla/2.0/relnotes/unix-2.0b5.html>

- Netscape Navigator 2.0b5 for Windows, Netscape Corp., <http://home.netscape.com/eng/mozilla/2.0/relnotes/windows-2.0b5.html>
- Netscape Navigator 2.0b6 for Windows, Netscape Corp., <http://home.netscape.com/eng/mozilla/2.0/relnotes/windows-2.0b6.html>
- Netscape Navigator 2.0b6a for Macintosh, Netscape Corp., <http://home.netscape.com/eng/mozilla/2.0/relnotes/mac-2.0b6a.html>
- Netscape Navigator 2.0b6a for Unix, Netscape Corp., <http://home.netscape.com/eng/mozilla/2.0/relnotes/unix-2.0b5.html>
- Netscape Navigator 2.0b6a for Unix, Netscape Corp., <http://home.netscape.com/eng/mozilla/2.0/relnotes/unix-2.0.html>
- Netscape Navigator Handbook, Netscape Corp., 1/1996, <http://home.netscape.com/eng/mozilla/2.0/handbook/> (with linked materials)
- Netscape Release Notes, Netscape Corp., 1997, <http://home.netscape.com/eng/mozilla/2.0/relnotes/>
- Netscape to License Sun's Java Programming Language, Netscape Corp., 5/1995, <http://home.netscape.com/newsref/pr/newsrelease25.html>
- Netscape Unveils Netscape Navigator 1.1, Netscape Corp., 3/1995, <http://home.netscape.com/newsref/pr/newsrelease16.html> (with linked materials)
- Netscape Unveils Netscape Navigator Personal Edition for Windows 95, Netscape Corp., 8/1995, <http://home.netscape.com/newsref/pr/newsrelease38.html>
- Netscape Unveils Netscape Navigator Personal Edition, Netscape Corp., 6/1995, <http://home.netscape.com/newsref/pr/newsrelease27.html>
- Netscape Unveils Windows 95 Version of Netscape Navigator, Netscape Corp., 6/1995, <http://home.netscape.com/newsref/pr/newsrelease30.html>
- Padmanabhan, Improving World Wide Web Latency, *University of California at Berkeley, Report No. UCB/CSD-95-875*, 1995, pp. 1-24.
- Pfaffenberger, Mosaic User's Guide, MIS:Press, 1994, pp. i-274
- Pitter et al., Every Student's Guide to the World Wide Web, McGraw-Hill, 1996, pp. i-146
- Randall, Teach Yourself the Internet: Around the World in 21 Days, *Sams Publishing*, 1994, pp. 212-227

- Schwerin, How to Use Netscape Navigator 2.0, Ziff-Davis Press, 1996, pp. i-219
- Shrikumar et al., Thinternet: life at the end of a tether, *Computer Networks and ISDN Sys.* 27, 1994, pp. 375-385
- Spyglass Mosaic (all versions available prior to 4/1996)
- Tannenbaum, Netscape Surfs The World Wide Web Better Than Mosaic, *Network Computing*, 1/1995, Vol. 6, No. 1, Issue 44, <http://www.networkcomputing.com/601/601sneak1.html>
- U.S. Patent No. 5,446,740 to Yien et al. (8/1995)
- U.S. Patent No. 5,493,677 to Balogh et al. (2/1996)
- U.S. Patent No. 5,495,565 to Millard et al. (2/1996)
- U.S. Patent No. 5,499,330 to Lucas et al. (3/1996)
- U.S. Patent No. 5,528,739 to Lucas et al. (6/1996)
- U.S. Patent No. 5,530,852 to Meske, Jr. et al. (6/1996)
- U.S. Patent No. 5,537,526 to Anderson et al. (7/1996)
- U.S. Patent No. 5,544,051 to Senn et al. (8/1996)
- U.S. Patent No. 5,572,643 to Judson (11/1996)
- U.S. Patent No. 5,621,874 to Lucas et al. (4/1997)
- U.S. Patent No. 5,623,652 to Vora et al. (4/1997)
- U.S. Patent No. 5,625,818 to Zarmer et al. (4/1997)
- U.S. Patent No. 5,649,186 to Ferguson (7/1997)
- U.S. Patent No. 5,657,390 to Elgamal et al. (8/1997)
- U.S. Patent No. 5,659,729 to Nielsen (8/1997)
- U.S. Patent No. 5,675,507 to Bobo, II (10/1997)
- U.S. Patent No. 5,701,451 to Rogers et al. (12/1997)
- U.S. Patent No. 5,710,918 to Lagarde et al. (1/1998)
- U.S. Patent No. 5,715,445 to Wolfe (2/1998)

- U.S. Patent No. 5,721,851 to Cline et al. (2/1998)
- U.S. Patent No. 5,724,514 to Arias (3/1998)
- U.S. Patent No. 5,737,599 to Rowe et al. (4/1998)
- U.S. Patent No. 5,745,360 to Leone et al. (4/1998)
- U.S. Patent No. 5,748,931 to Jones et al. (5/1998)
- U.S. Patent No. 5,761,683 to Logan et al. (6/1998)
- U.S. Patent No. 5,764,235 to Hunt et al. (6/1998)
- U.S. Patent No. 5,781,714 to Collins et al. (7/1998)
- U.S. Patent No. 5,781,785 to Rowe et al. (7/1998)
- U.S. Patent No. 5,805,815 to Hill (9/1998)
- U.S. Patent No. 5,815,663 to Uomini (9/1998)
- U.S. Patent No. 5,978,817 to Giannandrea et al. (11/1999)
- U.S. Patent No. 6,012,071 to Krishna et al. (1/2000)
- U.S. Patent No. 6,119,135 to Helfman (9/2000)
- U.S. Patent No. 6,487,588 to Phillips (11/2002)
- U.S. Patent No. 6,507,872 to Geshwind (1/2003)
- U.S. Patent No. 6,789,263 to Shimada et al. (9/2004)
- U.S. Patent No. 7,447,738 to Andrews et al. (11/2004)
- W3C's Arena web browser, including at least preBeta versions 0.93 through 0.98h, beta-1a through beta-2b, and all other versions available prior to 4/1996
- Web Graphics: Small and Quick Is In, *Infoworld*, 6/1995, p. 75
- Wei et al., SAGE: A HyperCard-Based GDSS, 1992 *IEEE*, pp. 14-22
- Welcome to Arena 0.98, W3C, <http://www.w3.org/Arena/0.98picture>
- Welcome to MacWeb, EINet, 3/1995 (with linked materials)
- Wiggins, Examining Mosaic, A History and Review, *Internet World*, 10/1995, pp. 48-51

B. Name and address of any person who may be relied upon as the prior inventor or as having prior knowledge of or as having previously used or offered for sale:

- Individuals formerly associated with Netscape Corp. (address unknown)
- Individuals formerly associated with Spyglass, Inc. (address unknown)
- Individuals formerly or currently associated with the World Wide Web Consortium (W3C) (address unknown)
- Individuals formerly or currently associated with the National Center for Supercomputing Applications (NCSA) (address unknown)

II. '522 Patent

A. Prior Art

- Anderson et al., Instantaneous Combination Boxes Using Presentation Window Parameters, *IBM TDB*, 6/1994, Vol. 37, No. 06A, pp. 317-318.
- Berry et al., The Evolution of the Common User Access Workplace Model, *IBM Sys. Journal*, Vol. 31, Issue 3, 1992, pp. 414-428
- Berry, The Designer's Model of the CUA Workspace, *IBM Sys. Journal*, Vol. 31, Issue 3, 1992, pp. 429-458
- Boyce, Maximizing Windows 3.1, New Riders Pub., 1992, pp. 1-800
- Brininstool et al., Graphical Notebook Control Tab Scroll Button Action Settings, *IBM TDB*, 9/1992, Vol. 35, No. 4B, pp. 371-372.
- Carr et al., The Power of Penpoint, Addison-Wesley Publishing Co., 2/1991, pp. i-342
- Carr et al., The Power of PenPoint: Chapters 1 and 2, Addison-Wesley: Reading, MA, 1991, <http://www.guidebookgallery.org/books/thepowerofpenpoint/chapter1> & <http://www.guidebookgallery.org/books/thepowerofpenpoint/chapter2>.
- Coulombe et al., Visual Two Level Information Categorization, *IBM TDB*, 7/1992, Vol. 35, No. 2, pp. 189-190.
- Cox et al., Object-Oriented Programming: An Evolutionary Approach, 1991.
- Cuthbertson et al., Optimizing Notebook Size to Reflect Dynamically Added Notebook Pages in a Distributed Environment, *IBM TDB*, 12/1994, Vol. 37, No. 12, pp. 227-228.

- DE 4236820 to Ricci (5/1994)
- EP 0336552 to Horn et al. (10/1989)
- Fitzpatrick et al., Notebook Control Cascading Tab Choices, *IBM TDB*, 9/1992, Vol. 35, No. 4B, pp. 284-286.
- Flowers et al., X Windows System, Digital Press, 3rd Ed., 1992, pp. i-914
- Guide to Programming, Microsoft^{RTM}, WindowsTM 3.1, Microsoft Press, Chapter 8, pp. 164-188, 1992.
- Haynes, Multi-dimensional Booktab Control for Graphical User Interfaces, *IBM TDB*, 9/1991, Vol. 34, No. 4A, pp. 208-210.
- IBM OS/2 2.0
- IBM OS/2 2.0 Technical Library, Programming Guide Volume I, Version 2.0, IBM Corp., 1st Ed., 3/1992
- IBM OS/2 2.0 Technical Library, Programming Guide Volume II, Version 2.0, IBM Corp., 1st Ed., 3/1992
- IBM OS/2 2.0 Technical Library, Programming Guide Volume III, Version 2.0, IBM Corp., 1st Ed., 3/1992
- IBM OS/2 Technical Library, OS/2 2.0 Application Design Guide, The Official Guide, QUE, 1st Ed. Mar. 1992
- IBM OS/2 Technical Library, OS/2 2.0 Control Program Programming Guide, Que, 3/1992, pp. i-17-6
- Jefferson, Presentation Manager CUA File Dialog Processor, *IBM TDB*, 9/1991, Vol. 34, No. 4B, pp. 83-84.
- Knight, Learning to Program OS/2 2.0 Presentation Manager by Example, Van Nostrand Reinhold Pub., 8/1992, pp. 1-230.
- Martin et al., System Application Architecture, Common User Access, Prentice-Hall, Inc., 1991, pp. i-337
- Miller, Using Norton Desktop 3 for Windows, Que, 1993, pp. 1-600
- Morgan et al., Backward Compatibility to Native Presentation Manager from an Extended Presentation Manager Architecture, *IBM TDB*, 3/1994, Vol. 37, No. 3, pp. 545-546.

- Object-Oriented Interface Design, IBM Common User Access™ Guidelines, Que, 12/1992, pp. i-668
- OS/2 2.0 IBM's Challenge to Windows, Computer Technology Research Corp., 7/1992, pp. i-69
- OS/2 2.0 Information and Planning Guide, Document No. G326-0160-00, IBM Corp., 1st Ed., 4/1992, pp. i-102
- OS/2 API, *see, e.g.*, <http://www.edm2.com/os2api/>
- PenPoint™ UI Design Guidelines, GO Corp., Rev. 0.5, 2/1991, pp. i-198.
- Redpath, System Notebook Visual Rendition, *IBM TDB*, 7/1992, Vol. 35, No. 2, pp. 225-226.
- Schilit et al., The PARCTAB Mobile Computing System, *Fourth Workshop on Workstation Operating Systems*, 10/1993.
- Schmucker, Object-Oriented Programming for the Macintosh, 1986
- Systems Application Architecture, Common User Access, Advanced Interface Design Guide, IBM Corp., 6/1989, pp. i-219
- U.S. Patent No. 5,119,475 to Smith et al. (6/1992)
- U.S. Patent No. 5,168,441 to Onarheim et al. (12/1992)
- U.S. Patent No. 5,237,651 to Randall (8/1993)
- U.S. Patent No. 5,283,864 to Knowlton (2/1994)
- U.S. Patent No. 5,305,435 to Bronson (4/1994)
- U.S. Patent No. 5,327,529 to Fults et al. (7/1994)
- U.S. Patent No. 5,347,295 to Agulnick et al. (9/1994)
- U.S. Patent No. 5,367,624 to Cooper (11/1994)
- U.S. Patent No. 5,379,432 to Orton et al. (1/1995)
- U.S. Patent No. 5,392,387 to Fitzpatrick et al. (2/1995)
- U.S. Patent No. 5,416,895 to Anderson et al. (5/1995)
- U.S. Patent No. 5,428,729 to Chang et al. (6/1995)

- U.S. Patent No. 5,463,724 to Anderson et al. (10/1995)
- U.S. Patent No. 5,500,929 to Dickinson (3/1996)
- U.S. Patent No. 5,502,805 to Anderson (3/1996)
- U.S. Patent No. 5,515,497 to Itri et al. (5/1996)
- U.S. Patent No. 5,530,865 to Owens et al. (6/1996)
- U.S. Patent No. 5,537,548 to Fin et al. (7/1996)
- U.S. Patent No. 5,542,040 to Chang et al. (7/1996)
- U.S. Patent No. 5,544,288 to Morgan et al. (8/1996)
- U.S. Patent No. 5,548,703 to Berry et al. (8/1996)
- U.S. Patent No. 5,550,563 to Matheny et al. (8/1996)
- U.S. Patent No. 5,557,732 to Thompson (9/1996)
- U.S. Patent No. 5,559,946 to Porter (9/1996)
- U.S. Patent No. 5,566,248 to Ulrich (10/1996)
- U.S. Patent No. 5,583,981 to Pleyer (10/1996)
- U.S. Patent No. 5,604,861 to Douglas et al. (2/1997)
- U.S. Patent No. 5,621,434 to Marsh (4/1997)
- U.S. Patent No. 5,623,591 to Cseri (4/1997)
- U.S. Patent No. 5,664,103 to Stein et al. (9/1997)
- U.S. Patent No. 5,673,315 to Wolf (9/1997)
- U.S. Patent No. 5,673,403 to Brown et al. (9/1997)
- U.S. Patent No. 5,721,847 to Johnson (2/1998)
- U.S. Patent No. 5,808,611 to Johnson et al. (9/1998)
- U.S. Patent No. 5,936,614 to An et al. (8/1999)
- Using the Operating System, IBM OS/2 Version 2.1, IBM Corp., 5/1993, pp i-514

- Voss, Object-Oriented Programming: An Introduction, 1991
- Weber, Consistent Dynamic Dialog Box Fields, *IBM TDB*, 7/1991, Vol. 34, No. 2, pp. 76-77.

B. Name and address of any person who may be relied upon as the prior inventor or as having prior knowledge of or as having previously used or offered for sale:

- Individuals formerly or currently associated with IBM Corp. (address unknown)

III. '780 Patent

A. Prior Art

- Albers et al., The Audible Web: Auditory Enhancements for Mosaic, *Chi '95 Mosaic of Creativity*, 5/1995, pp. 318-319
- Bartlett, Experience with a Wireless World Wide Web Client, Western Research Laboratory, WRL Technical Note TN-46, 3/1995, pp. 1-13
- Brown et al., A New Paradigm for Browsing the Web, *Conference companion on Human factors in computing systems*, 5/1995, pp. 320-321.
- Brown et al., DeckScape: An Experimental Web Browser, <http://www.cs.princeton.edu/~ras/deckscape/paper.html>
- Brown et al., The DeckScape Web Browser, *CHI 96 Electronic Proceedings*, http://www.sigchi.org/chi96/proceedings/videos/Brown/mhb_bdy.htm
- Carr et al., The Power of Penpoint, Addison-Wesley Publishing Co., 2/1991, pp. i-342
- Carts, How to put the World Wide Web to work for you, *Laser Focus World*, 1/1995, pp. 116-117
- Cheek, Web tools: join those making the Internet work for their agency's mission, *Government Computing News*, Vol. 15, No. 5, 3/1996, pp. 1-38
- Cook, Write a Simple HTTP-based Server Using MFC and Windows Sockets, *Microsoft Systems Journal*, 2/1996, pp. 1-29
- Cooper et al., PDA Web Browsers: Implementation Issues, <http://alethea.ukc.ac.uk/Dept/Computing/Research/STEP/Papers/WWWPDA/>, 11/1995, pp. 1-12

- deLespinasse, Rover Mosaic: E-Mail Communcation for a Full-Function Web Browser, submitted to the Dept. of Elec. Eng'g & Comp. Sci at the Mass. Institute of Tech., 6/1995, pp. 1-43.
- Finnie, Microsoft Internet Explorer 2.0, Microsoft Narrows Netscape's Lead, *PC Computing*, Vol. 9, No. 4, 4/1996, p. 79
- Fleming, Improving the Performance of the World Wide Web over Wireless Networks, submitted to the faculty of the Va. Polytechnic Inst. and State Univ., 11/1996, pp. i-34
- Gerber, Browsers on the Wild, Wild Web, *Network Computing*, 4/1995, p. 54 et seq.
- Gessler et al, PDAs as mobile WWW browsers, *Computer Networks and ISDN Systems* 28, 1995, pp. 53-59
- Greengard, Home, Home on the Web, *Personnel Journal*, Vol. 75, No. 3, 3/1996, pp. 1-9
- Herbst, Wefest IV, A Report of the Happenings at the Fourth World-Wide Web Conference, *Internet World*, 3/1996, pp. 22-28
- Hu et al., Parameterizable fonts based on shape components, *IEEE Computer Graphics and Applications*, Vol. 21, Issue 3, May/Jun. 2001, pp. 70-85.*
- Hypercard Ver. 1.2
- HyperTIES, e.g., <http://www.cs.umd.edu/hcil/hyperties/>;
- Ibrahim, World-Wide algorithm animation, *Computer Networks and ISDN Systems*, 1994, Vol. 27, Issue 2, pp. 255-265.
- Internet Explorer 1.0 and 2.0 for Windows, along with all other versions available prior to 5/1997.
- Jefferson et al., Presentation Manager Progress Indicator, *IBM TDB*, 7/1991, Vol. 34, No. 2, pp. 475-476.
- JP 7325807 to Kizaki (12/1995)
- JP 8221401 to Yamamoto (8/1996)
- Kaashoek et al., Dynamic Documents: Mobile Wireless Access to the WWW; *Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications*, 12/1994, pp. 1-6
- Kane, Prodigy Interactive Personal Service Made Easy, Osborne McGraw-Hill, 1991, pp. i-251
- Lauff et al., Multimedia Client Implemented on Personal Digital Assistants, IDMS, 1997

- Lavery, Internet Primer: Workshop Design and Objectives, *Internet Reference Services Quarterly*, 1996, Vol. 1(3), pp. 35-53.
- Liljeberg et al., Optimizing World-Wide Web for Weakly Connected Mobile Workstations: An Indirect Approach, *Proceedings of SDNE '95*, 6/1995, pp. 1-8
- Liu et al., Web-based peer review: the learner as both adapter and reviewer, *Education, IEEE Transactions*, Vol. 44, Issue 3, Aug. 2001, pp. 246-251.
- LunaSuite Pro (all versions available prior to 5/1997)
- MacWeb version 1.1.1E (*see, e.g.*, <http://www.youtube.com/watch?v=blywSnj4lgk>) along with all other versions available prior to 5/1996.
- Morris, HTML for Fun and Profit, SunSoft Press, 1995, pp. 63-72
- NCSA Mosaic (all versions available prior to 5/1997)
- NetHopper (all versions available prior to 5/1997)
- NetHopper Version 3.0 User's Manual, Allpen Software Inc., 1997, pp. 1-20
- Netscape Navigator versions 1.0, 1.1, 1.2, 1.22 and 2.0 for Windows, Macintosh, and/or Unix, along with all other versions available prior to 5/1997.
- Newt's Cape (all versions available prior to 5/1997)
- Newton MessagePad 2000 Tech Article, Apple Computer, Inc., 1/1997, pp. i-38
- PenPoint™ UI Design Guidelines, GO Corp., Rev. 0.5, 2/1991, pp. i-198.
- PocketWeb (all versions available prior to 5/1997)
- PocketWeb WWW Browser for the Apple Newton Message Pad, TecO, available at <http://www.teco.edu/pocketweb>
- Robertson et al., Using the World Wide Web to Provide a Platform Independent Interface to High Performance Computing, *Technologies for the Information Superhighway*, 1995, pp. 1-5.
- Romano, Jr. et al., User Driven Design of a Web-Based Group Support System, *Proceedings of the Thirtieth Annual Hawaii International Conference on Systems Sciences*, 1997, Vol. 2, pp. 1-10.
- Schilit et al., TeleWeb: Loosely Connected Access to the World Wide Web, *5th Int'l WWW Conf.*, 5/1996, pp. 1-14

- Smallman et al., Information availability in 2D and 3D displays, *IEEE Computer Graphics and Applications*, Vol. 21, Issue 5, Sep./Oct. 2001, pp. 51-57
- Spyglass Mosaic (all versions available prior to 5/1997)
- U.S. Patent No. 4,266,253 to Matherat (5/1981)
- U.S. Patent No. 5,301,348 to Jaaskelainen (4/1994)
- U.S. Patent No. 5,467,459 to Alexander et al. (11/1995)
- U.S. Patent No. 5,479,599 to Rockwell et al. (12/1995)
- U.S. Patent No. 5,572,643 to Judson (11/1996)
- U.S. Patent No. 5,630,148 to Norris (5/1997)
- U.S. Patent No. 5,712,654 to Kawashima et al. (1/1998)
- U.S. Patent No. 5,715,416 to Baker (2/1998)
- U.S. Patent No. 5,724,514 to Arias (3/1998)
- U.S. Patent No. 5,731,813 to O'Rourke et al. (3/1998)
- U.S. Patent No. 5,737,599 to Rowe et al. (4/1998)
- U.S. Patent No. 5,760,771 to Blonder et al. (6/1998)
- U.S. Patent No. 5,774,666 to Portuesi (6/1998)
- U.S. Patent No. 5,799,267 to Siegel (8/1998)
- U.S. Patent No. 5,805,166 to Hall, Jr. et al. (9/1998)
- U.S. Patent No. 5,805,815 to Hill (9/1998)
- U.S. Patent No. 5,809,242 to Shaw et al. (9/1998)
- U.S. Patent No. 5,845,282 to Alley et al. (12/1998)
- U.S. Patent No. 5,864,850 to Nordman (1/1999)
- U.S. Patent No. 5,877,766 to Bates et al. (3/1999)
- U.S. Patent No. 5,907,843 to Cleron et al. (5/1999)
- U.S. Patent No. 5,908,467 to Barrett et al. (6/1999)

- U.S. Patent No. 5,956,509 to Kevner (9/1999)
- U.S. Patent No. 5,969,705 to Fisher et al. (10/1999)
- U.S. Patent No. 5,973,692 to Knowlton et al. (10/1999)
- U.S. Patent No. 5,983,005 to Monteiro et al. (11/1999)
- U.S. Patent No. 6,023,698 to Lavey, Jr. et al. (2/2000)
- U.S. Patent No. 6,049,812 to Bertram et al. (4/2000)
- U.S. Patent No. 6,065,059 to Shieh et al. (5/2000)
- U.S. Patent No. 6,097,390 to Marks (8/2000)
- U.S. Patent No. 6,101,509 to Hanson et al. (8/2000)
- U.S. Patent No. 6,101,510 to Stone et al. (8/2000)
- U.S. Patent No. 6,182,072 to Leak et al. (1/2001)
- U.S. Patent No. 6,223,188 to Albers et al. (4/2001)
- U.S. Patent No. 6,266,082 to Yonezawa et al. (7/2001)
- U.S. Patent No. 6,401,099 to Koppolu et al. (6/2002)
- U.S. Patent No. 6,584,498 to Nguyen (6/2003)
- U.S. Patent No. 6,615,251 to Klug et al. (9/2003)
- Web Graphics: Small and Quick Is In, *Infoworld*, 6/1995, p. 75
- Wiggins, Examining Mosaic, A History and Review, *Internet World*, 10/1995, pp. 48-51
- www.sciam.com/200/1100issue/1100stjohnbox1.html

IV. **'551 Patent**

A. Prior Art

- Acorn A5000 Welcome Guide, Acorn Computers Ltd., pp. 1-88
- Adobe Acrobat 3.0 Reader Online Guide, Adobe Systems, Inc., pp. 1-110

- aha! 2.0 for Windows InkWriter.TM. The power of word processing, the convenience of pen and paper, Advertisement, 1994, 1995, 2 pp., aha! software corporation, Mountain View, CA
- aha! InkWriter.TM. for Magic Cap The fastest, most convenient way to create faxes, notes and messages. Advertisement, 1994, 1995, 2 pp., aha! software corporation, Mountain View, CA
- aha! InkWriter.TM. The simplicity of pen and paper, the power of word processing., Advertisement, 1983, 2 pp., aha! software corporation, Mountain View, CA
- aha! software products, Web Page List of Products, Sep. 24, 1997, pp. 1-5, aha! software corporation, Mountain View, CA
- Carr et al., The Power of Penpoint, Addison-Wesley Publishing Co., 2/1991, pp. i-342
- Chiu et al., A Dynamic Grouping Technique for Ink and Audio Notes, *Proceedings of the 11th annual ACM symposium on User interface software and technology*, 1998, pp. 195-202.
- Chiu et al., A Dynamic Grouping Technique for Ink and Audio Notes, *UIST '98 San Francisco, CA*, 1998, pp. 195-202
- Chuah et al., SDM: Selective Dynamic Manipulation of Visualizations, *UIST 95 Pittsburgh PA*, 11/1995, pp. 61-70.
- Computer Networks and ISDN Systems, Responsive interaction for a large Web application: the meteor shower architecture in the WebWriter II Editor, 1997, pp. 1508-1517
- Dublin Core Metadata Element Set, Version 1.1: Reference Description 1999.
- Dublin Core Metadata Initiative: User Guide Working Draft Jul. 31, 1998 1999.
- Dublin Core Resource Types: Structurality DRAFT: Jul. 24, 1997, 2/1998
- EP 0 902 379 to Schilit et al. (3/1999)
- EP 1 016 983 to Schilit et al. (7/2000)
- Explore the Features of the Rocket eBook;
wysiwyg://212http://www.rocketbook.com/Products/Tour/index.html
- Gillan et al., How Does Fitts' Law Fit Pointing and Dragging? *CHI '90 Proceedings of the SIGCHI conference on Human factors in computing systems: Empowering people*, 4/1990, pp. 227-234.

- Golovchinsky et al., From Reading to Retrieval: Freeform Ink Annotations as Queries, *SIGIR '99: Proceedings of the 22nd Annual Int'l ACM Sig. Conf. on R&D in Information Retrieval*, 8/1999, pp. 9-25
- Hirotsu et al., Cmew/U--A Multimedia Web Annotation Sharing System, NTT Network Innovation Laboratories--Japan, IEEE, 1999, pp. 356-359
- Hudson, User Interface Specification Using an Enhanced Spreadsheet Model, *ACM Transactions on Graphics*, 1994, Vol. 13, Issue 3, pp. 1-25.
- Igarashi et al., An Architecture for Pen-Based Interaction of Electronic Whiteboards, 2000
- JP 7006173 to Kinoshita (1/1995)
- JP 7021159 to Maruoka (1/1995)
- Kankaanpaa, FIDS—A Flat-Panel Interactive Display System, *1988 IEEE Computer Graphics & Applications*, pp. 71-82, 3/1998
- Kay, A Personal Computer for Children of All Ages, 8/1972, pp. 1-11
- KR 20000036836 to Song et al. (7/2000)
- KR 20000060953 to Lim (10/2000)
- Kristensen, A., Formsheets and the XML Forms Language, 3/1999
- Kunikazu, T., Patent Abstracts of Japan, Publication No. 11327789 for Color Display and Electronic Blackboard System, published Nov. 30, 1999
- Kuo et al., A Synchronization Scheme for Multimedia Annotation, Multimedia Information Networking Laboratory, Tamkang University--Taiwan, IEEE, 1997, pp. 594-598
- Lee et al., A Multi-Touch Three Dimensional Touch-Sensitive Tablet, *CHI '85 Proceedings*, 4/1985, pp. 21-25
- Leopold et al., A User Interface for the Visualization and Manipulation of Arrays, *IEEE Symposium on Visual Languages*, 1996, pp. 54-55
- Leopold et al., Keyboardless Visual Programming Using Voice, Handwriting, and Gesture, *IEEE Symposium on Visual Languages*, 1997, pp. 28-35.
- Marshall et al., Spatial Hypertext and the Practice of Information Triage, *Hypertext 97*, 1997, pp. 124-134

- Marshall et al., Spatial Hypertext: Designing for Change, *Communications of the ACM*, Vol. 38, No. 8, 8/1995, pp. 88-97
- Mel et al., Tablet: The Personal Computer of the Year 2000, *Communications of the ACM*, 6/1988, pp. 638-646
- Moran, Pen-Based Interaction Techniques for Organizing Material on an Electronic Whiteboard, *UIST '97 Proceedings of the 10th annual ACM symposium on User interface software and technology*, 1997
- Myers et al., Marquise: Creating Complete User Interfaces by Demonstration, *CHI '93 Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems*, 4/1993, pp. 293-300.
- Myers, Gilt Reference Manual: A Simple Interface Builder for Garnet, Carnegie Mellon University, 10/1993, pp. 515-532.
- Open eBook Forum, Open eBook.TM. Publication Structure 1.0, Sep. 1999
<http://www.ebxwg.org/oebps/oebps1.0/download/oeb1-oebps.htm>
- PenPoint™ UI Design Guidelines, GO Corp., Rev. 0.5, 2/1991, pp. i-198.
- Raisamo, Evaluating Different Touch-Based Interaction Techniques in a Public Information Kiosk, *Conference of the CHI Special Interest Group of the Ergonomics Society of Australia*, 1999, pp. 1-11.
- Ren et al., Improving Selection Performance on Pen-Based Systems: A Study of Pen-Based Interaction for Selection Tasks, *ACM Transactions on Computer-Human Interaction*, 9/2000, pp. 384-416.
- Rocket eBook Using the Rocket eBook, 1999;
<http://www.rocket.com/Products/Faq/using.html>.
- Rubine, The Automatic Recognition of Gestures, Submitted to the Dept. of Computer Science at Carnegie Mellon Univ., 12/1991, pp. i-265
- Schilit et al., Beyond Paper: Supporting Active Reading with Free Form Digital Ink Annotations, *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1998, pp. 1-8.
- Schilit et al., Digital Library Information Appliances, Association for Computing Machinery, XP000889731, Jun. 23, 1998, e-mail: (Schilit,price,gene) @pel.xerox.com. pp. 217-226.
- SOFTBOOK™. Press, The Leader in Internet-Enabled Document Distribution and Reading Systems: 1999 <http://www.softbook.com/consumer/reader.asp>.

- Stallman, GNU Emacs Manual, 7/1994, pp. 1 et seq.
- Text Editing, *VP Series Reference Library*, Version 1.0, Xerox Corporation, 1985, pp. 47-56
- The RISC OS Drag-and-Drop System, Acorn Computers Ltd., 10/1993, pp. 1-7.
- The Windows Interface Guidelines — A Guide for Designing Software, Microsoft Corp., 2/1995, available at http://www.ics.uci.edu/~kobsa/courses/ICS104/course-notes/Microsoft_WindowsGuidelines.pdf
- U.S. Patent No. 4,924,433 to Hirata et al. (5/1990)
- U.S. Patent No. 5,070,478 to Abbott (12/1991)
- U.S. Patent No. 5,146,552 to Cassorla et al. (9/1992)
- U.S. Patent No. 5,239,466 to Morgan et al. (8/1993)
- U.S. Patent No. 5,263,134 to Paal et al. (11/1993)
- U.S. Patent No. 5,347,295 to Agulnick et al. (9/1994)
- U.S. Patent No. 5,367,625 to Ishitani (11/1994)
- U.S. Patent No. 5,396,590 to Kreegar (3/1995)
- U.S. Patent No. 5,428,805 to Morgan (6/1995)
- U.S. Patent No. 5,434,965 to Matheny et al. (7/1995)
- U.S. Patent No. 5,442,742 to Greyson et al. (8/1995)
- U.S. Patent No. 5,473,742 to Polyakov et al. (12/1995)
- U.S. Patent No. 5,487,141 to Cain et al. (1/1996)
- U.S. Patent No. 5,513,309 to Meier et al. (4/1996)
- U.S. Patent No. 5,517,578 to Altman et al. (5/1996)
- U.S. Patent No. 5,523,775 to Capps (6/1996)
- U.S. Patent No. 5,544,048 to Dugan et al. (8/1996)
- U.S. Patent No. 5,551,148 to Wellner (4/1996)
- U.S. Patent No. 5,572,643 to Judson (11/1996)

- U.S. Patent No. 5,574,840 to Kwatinetz et al. (11/1996)
- U.S. Patent No. 5,613,019 to Altman et al. (3/1997)
- U.S. Patent No. 5,617,528 to Stechmann et al. (4/1997)
- U.S. Patent No. 5,625,833 to Levine et al. (4/1997)
- U.S. Patent No. 5,630,040 to Furuya (5/1997)
- U.S. Patent No. 5,632,022 to Warren et al. (5/1997)
- U.S. Patent No. 5,663,748 to Huffman et al. (9/1997)
- U.S. Patent No. 5,664,210 to Fleming et al. (9/1997)
- U.S. Patent No. 5,666,552 to Greyson et al. (9/1997)
- U.S. Patent No. 5,680,636 to Levine et al. (10/1997)
- U.S. Patent No. 5,682,439 to Beernink et al. (10/1997)
- U.S. Patent No. 5,687,331 to Volk et al. (11/1997)
- U.S. Patent No. 5,717,879 to Moran et al. (2/1998)
- U.S. Patent No. 5,727,129 to Barrett et al. (3/1998)
- U.S. Patent No. 5,729,255 to Aoki (3/1998)
- U.S. Patent No. 5,729,673 to Cooper et al. (3/1998)
- U.S. Patent No. 5,729,687 to Rothrock et al. (3/1998)
- U.S. Patent No. 5,737,599 to Rowe et al. (4/1998)
- U.S. Patent No. 5,737,618 to Nakayama et al. (4/1998)
- U.S. Patent No. 5,745,116 to Pisutha-Arnold (4/1998)
- U.S. Patent No. 5,751,283 to Smith (2/1998)
- U.S. Patent No. 5,754,873 to Nolan (5/1998)
- U.S. Patent No. 5,760,773 to Berman et al. (6/1998)
- U.S. Patent No. 5,761,682 to Huffman et al. (6/1998)

- U.S. Patent No. 5,790,818 to Martin (8/1998)
- U.S. Patent No. 5,801,687 to Peterson et al. (9/1998)
- U.S. Patent No. 5,815,595 to Gugler (9/1998)
- U.S. Patent No. 5,821,925 to Carey et al. (10/1998)
- U.S. Patent No. 5,832,263 to Hansen et al. (11/1998)
- U.S. Patent No. 5,832,528 to Kwatinetz et al. (11/1998)
- U.S. Patent No. 5,835,919 to Stern et al. (11/1998)
- U.S. Patent No. 5,838,914 to Carleton et al. (11/1998)
- U.S. Patent No. 5,845,262 to Nozue et al. (12/1998)
- U.S. Patent No. 5,867,144 to Wyard (2/1999)
- U.S. Patent No. 5,877,757 to Baldwin et al. (3/1999)
- U.S. Patent No. 5,889,523 to Wilcox et al. (3/1999)
- U.S. Patent No. 5,893,126 to Drews et al. (4/1999)
- U.S. Patent No. 5,893,132 to Huffman et al. (4/1999)
- U.S. Patent No. 5,918,236 to Wical (6/1999)
- U.S. Patent No. 5,920,694 to Carleton et al. (7/1999)
- U.S. Patent No. 5,930,813 to Padgett et al. (7/1999)
- U.S. Patent No. 5,931,912 to Wu et al. (8/1999)
- U.S. Patent No. 5,933,140 to Strahorn et al. (8/1999)
- U.S. Patent No. 5,937,416 to Menzel (8/1999)
- U.S. Patent No. 5,948,040 to DeLorme et al. (9/1999)
- U.S. Patent No. 5,950,214 to Rivette et al. (9/1999)
- U.S. Patent No. 5,960,114 to Dauerer et al. (9/1999)
- U.S. Patent No. 5,973,692 to Knowlton et al. (10/1999)

- U.S. Patent No. 5,978,818 to Lin (11/1999)
- U.S. Patent No. 5,983,248 to DeRose et al. (11/1999)
- U.S. Patent No. 5,986,665 to Wrey et al. (11/1999)
- U.S. Patent No. 6,011,537 to Slotznick (1/2000)
- U.S. Patent No. 6,012,055 to Campbell et al. (1/2000)
- U.S. Patent No. 6,018,334 to Eckerberg et al. (1/2000)
- U.S. Patent No. 6,018,342 to Bristor (1/2000)
- U.S. Patent No. 6,018,742 to Herbert, III (1/2000)
- U.S. Patent No. 6,035,330 to Astiz et al. (3/2000)
- U.S. Patent No. 6,049,812 to Bertram et al. (4/2000)
- U.S. Patent No. 6,052,514 to Gill et al. (4/2000)
- U.S. Patent No. 6,054,990 to Tran (4/2000)
- U.S. Patent No. 6,081,829 to Sidana (6/2000)
- U.S. Patent No. 6,088,481 to Okamoto et al. (7/2000)
- U.S. Patent No. 6,091,930 to Mortimer et al. (7/2000)
- U.S. Patent No. 6,094,197 to Buxton et al. (7/2000)
- U.S. Patent No. 6,118,437 to Fleck et al. (9/2000)
- U.S. Patent No. 6,122,649 to Kanerva et al. (9/2000)
- U.S. Patent No. 6,128,007 to Seybold (10/2000)
- U.S. Patent No. 6,151,426 to Lee et al. (11/2000)
- U.S. Patent No. 6,157,381 to Bates et al. (12/2000)
- U.S. Patent No. 6,178,243 to Pomerantz et al. (1/2001)
- U.S. Patent No. 6,181,344 to Tarpenning et al. (1/2001)
- U.S. Patent No. 6,195,679 to Bauersfeld et al. (2/2001)

- U.S. Patent No. 6,195,694 to Chen et al. (2/2001)
- U.S. Patent No. 6,199,082 to Ferrel et al. (3/2001)
- U.S. Patent No. 6,205,455 to Umen et al. (3/2001)
- U.S. Patent No. 6,230,171 to Pacifici et al. (5/2001)
- U.S. Patent No. 6,239,792 to Yanagisawa et al. (5/2001)
- U.S. Patent No. 6,262,719 to Bi et al. (7/2001)
- U.S. Patent No. 6,271,840 to Finseth et al. (8/2001)
- U.S. Patent No. 6,279,005 to Zellweger (8/2001)
- U.S. Patent No. 6,279,014 to Schilit et al. (8/2001)
- U.S. Patent No. 6,289,362 to Van Der Meer (9/2001)
- U.S. Patent No. 6,301,590 to Siow et al. (10/2001)
- U.S. Patent No. 6,321,244 to Liu et al. (11/2001)
- U.S. Patent No. 6,331,867 to Eberhard et al. (12/2001)
- U.S. Patent No. 6,335,727 to Morishita et al. (1/2002)
- U.S. Patent No. 6,340,967 to Maxted (1/2002)
- U.S. Patent No. 6,356,287 to Ruberry et al. (3/2002)
- U.S. Patent No. 6,369,811 to Graham et al. (4/2002)
- U.S. Patent No. 6,389,434 to Rivette et al. (5/2002)
- U.S. Patent No. 6,393,422 to Wone (5/2002)
- U.S. Patent No. 6,405,221 to Levine et al. (6/2002)
- U.S. Patent No. 6,421,065 to Walden et al. (7/2002)
- U.S. Patent No. 6,426,760 to Takahashi (7/2002)
- U.S. Patent No. 6,446,110 to Lection et al. (9/2002)
- U.S. Patent No. 6,460,058 to Koppolu et al. (10/2002)

- U.S. Patent No. 6,486,895 to Robertson et al. (11/2002)
- U.S. Patent No. 6,490,603 to Keenan et al. (12/2002)
- U.S. Patent No. 6,493,006 to Gourdol et al. (12/2002)
- U.S. Patent No. 6,525,749 to Moran et al. (2/2003)
- U.S. Patent No. 6,535,294 to Arledge, Jr. et al. (3/2003)
- U.S. Patent No. 6,539,370 to Chang et al. (3/2003)
- U.S. Patent No. 6,549,220 to Hsu et al. (4/2003)
- U.S. Patent No. 6,560,621 to Barile (5/2003)
- U.S. Patent No. 6,571,295 to Sidana (5/2003)
- U.S. Patent No. 6,610,102 to Aldred et al. (8/2003)
- U.S. Patent No. 6,628,285 to Abeyta et al. (9/2003)
- U.S. Patent No. 6,640,010 to Seeger et al. (10/2003)
- U.S. Patent No. 6,681,046 to Kashiwagi et al. (1/2004)
- U.S. Patent No. 6,683,631 to Carroll (1/2004)
- U.S. Patent No. 6,707,474 to Beck et al. (3/2004)
- U.S. Patent No. 6,848,082 to Patel et al. (12/1999)
- U.S. Patent No. 6,965,454 to Silverbrook et al. (11/2005)
- U.S. Patent No. 7,032,171 to Carroll (4/2006)
- U.S. Patent No. 7,191,405 to Jaramillo (3/2007)
- U.S. Patent No. 7,197,718 to Westerman et al. (3/2007)
- U.S. Patent No. 7,428,701 to Gavin et al. (9/2008)
- U.S. Patent Pub. 2003/0103082 to Carroll (6/2003)
- U.S. Patent Pub. 2004/0085368 to Johnson, Jr. et al. (5/2004)
- U.S. Patent Pub. 2006/0129933 to Land et al. (6/2006)

- Westerman, Handtracking, Finger Identification, and Chordic Manipulation on a Multi-Touch Surface, Submitted to the Univ. of Delaware, 1999, pp. i-333
- WO 00/75766 A1 to MaCinnes (12/2000)
- WO 87/01481 to Stein (3/1998)
- WO 96/20908 (7/1996)
- WO 97/22109 to Huffman et al. (6/1997)
- WO 97/36225 A1 to Gillespie et al. (10/1997)
- WO 98/06054 to Huffman et al. (2/1998)
- WO 98/09446 to Ho (3/1998)
- WO 99/49383 to Ruberry et al. (9/1999)

V. '233 Patent

A. Prior Art

- Acrobat Distiller Parameters, Adobe Developer Support, Technical Note #5151, Adobe Systems Inc., 9/1996, pp. 1-24
- Add or Remove Tools, Ambia Corp., <http://www.ambia.com/options.htm>, 1996
- Adkins et al., GSS Collaboration in Document Development: Using GroupWriter to Improve the Process, *Proceedings of the 32nd Hawaii International Conference on System Sciences*, 3/1999, pp. 1-11.
- Adobe Acrobat 3.0 Reader Online Guide, Adobe Systems, Inc., pp. 1-110.
- Adobe® Acrobat® 3.0 Tutorials, Adobe Systems Inc., 1996
- Adobe® Acrobat® 3.0 Online Guide, Adobe Systems Inc., 1996
- Adobe® Acrobat® 3.0 Online Guide, Adobe Systems Inc., 1996
- Adobe® Acrobat® Capture Plug-In Online Guide, Adobe Systems Inc., 1996
- Adobe® Acrobat® Catalog Online Guide, Adobe Systems Inc., 1996
- Adobe® Acrobat® Distiller Online Guide, Adobe Systems Inc., 1996

- Adobe® Acrobat® Exchange 3.0
- Adobe® Acrobat® Exchange 3.0 Online Guide, Adobe Systems Inc., 1996
- Adobe® Acrobat® PDF Writer Online Guide, Adobe Systems Inc., 1996
- Adobe® Acrobat® Reader® CD-ROM Online Guide, Adobe Systems Inc., 1996
- Adobe® Acrobat® Scan Online Guide, Adobe Systems Inc., 1996
- AdobeSM Customer Services, Guidelines for Using Adobe Acrobat 3.0 for Production Printing, Adobe Systems Inc., 1996, pp. 1-2
- Aha! 2.0 for Windows InkWriter.quadrature. The power of word processing, the convenience of pen and paper, Advertisement, 1994, 1995, 2 pp., aha! software corporation, Mountain View, CA.
- Aha! InkWriter.quadrature. for Magic Cap The fastest, most convenient way to create faxes, notes and messages. Advertisement, 1993, 1994, 2 pp., aha! software corporation, Mountain View, CA.
- Aha! InkWriter.quadrature. The simplicity of pen and paper, the power of word processing., Advertisement, 1993, 2 pp., aha! software corporation, Mountain View, CA.
- Aha! software products, Web Page List of Products, 9/1997, pp. 1-5, aha! software corporation, Mountain View, CA.
- Annotation Filter, Ambia Corp., <http://www.ambia.com/filter.htm>, 1996
- Annotation Manager, Ambia Corp., <http://www.ambia.com/anno-man.htm>, 1996
- Bloomer, WinLab Reviews, Aerial 1.1 and Remark 2.0, Add-Ons Make Acrobat Do Backflips, CMP Media Inc., *Windows Magazine*, 3/1997, p. 152, <http://www.winmag.com/library/1997/0301/winla124.htm>
- Cassorla et al., Onscreen Possibilities ... Today!, *1986 Intl. Professional Communication Conference, IEEE*, 10/1986, pp. 11-15
- Catlin et al., InterNote: Extending a Hypermedia Framework to Support Annotative Collaboration, *Proceedings of the second annual ACM conference on Hypertext*, 11/1989, pp. 365-378.
- Computer Networks and ISDN Systems Responsive interaction for a large Web application: the meteor shower architecture in the WebWriter II Editor; 1997, pp. 1508-1517.
- Configuration Manager, Ambia Corp., <http://www.ambia.com/config-man.htm>, 1996

- Crespo, Chang, and Bier, Responsive interaction for a large Web application: the meteor shower architecture in the WebWriter II Editor, *Computer Networks and ISDN Systems*, pp. 1508-1517.
- Davis et al., Shared Annotation for Cooperative Learning. *The first international conference on Computer support for collaborative learning*, 10/1995, pp. 84-88.
- Delete All Re:mark Annotations, Ambia Corp., <http://www.ambia.com/delete.htm>, 1996
- DeRose, XML Linking, *ACM Computing Surveys*, 12/1999, Vol. 31, Issue 4es, pp. 1-12.
- Dublin Core Metadata Element Set, Version 1.1: Reference Description 1999.
- Dublin Core Metadata Initiative: User Guide Working Draft Jul. 31, 1998, 1999.
- Dublin Core Resource Types: Structurality DRAFT: Jul. 24, 1997, 2/1998.
- Eisen et al., Searching for Annotation in Documents On a Computer, *IBM TDB*, 10/1/1990, Research Disclosure n318 10-90.
- Elsen et al., Searching for Annotaiton in Documents on a Computer, *IBM Tech. Disclosure Bulletin*, 10/1990
- Elsen et al., Using a Highlighter on Shared computer Documents, *IBM Tech. Disclosure Bulletin*, 10/1990
- EP 0 902 379 (A2) to Schilit et al. (3/1999)
- EP App. 0 650 126 A1 to Albini et al. (4/1995)
- Explore the Features of the Rocket eBook;
[wysiwyg://212http://www.rocketbook.com/Products/Tour/index.html](http://www.rocketbook.com/Products/Tour/index.html).
- Fielding et al., Web-Based Development of Complex Information Products, *Communications of the ACM*, 8/1998, Vol. 41, Issue 8, pp. 84-92.
- Flurry et al., Handwritten Annotation of Electronic Documents, *IBM TDB*, 7/1/1983, 07-83, pp. 504-507.
- Getting Started Adobe® Acrobat® version 3.0, Adobe Systems Inc., 1996, pp. i-30
- Hirotsu et al., Cmew/U--A Multimedia Web Annotation Sharing System, NTT Network Innovation Laboratories--Japan, IEEE, 1999, pp. 356-359.
- IBM BookManager R3 Displaying Online Books, IBM Corp., Release 3, 9/1994, pp. i-260
- IBM BookManager R3 Getting Started, IBM Corp., Release 3, 9/1994, pp. i-42

- Ide, Corpus Encoding Standard: SGML Guidelines for Encoding Linguistic Corpora, *First International Language Resources and Evaluation Conference*, 1998.
- Import Annotations, Ambia Corp., <http://www.ambia.com/import.htm>, 1996
- JP 11175569 (A) to Ginsburg et al. (7/1999)
- Kristensen, A., Formsheets and the XML Forms Language, 3/1999.
- Kunikazu, T., Patent Abstracts of Japan, Publication No. 11327789 for Color Display and Electronic Blackboard System, 11/1999.
- Kuo et al., A Synchronization Scheme for Multimedia Annotation, Multimedia Information Networking Laboratory, Tamkang University--Taiwan, IEEE, 1997, pp. 594-598.
- Landow, Intermedia: An Introduction, available at <http://www.cyberartsweb.org/cpace/ht/HTatBrown/Intermedia.html>
- Landow, InterNote: Intermedia's hypertext collaborative work tool, available at http://www.cyberartsweb.org/cpace/ht/HTatBrown/InterNote_378.html
- Marais, Supporting Cooperative and Personal Surfing with a Desktop Assistant, 11/1987
- Mendelson, Acrobat Paper Clips and Pushpins, *PC Magazine*, 3/1997, <http://www.pcmag.com/issues/1606/pcmg0015.htm>
- Mixed Object Document Content Architecture Reference: Chapter 3, available at http://www.afpcinc.org/mo_dca_reference.pdf (accessed April 27, 2011), 1990, pp. 64-70.
- Neuwirth, The Notes Program: A Hypertext Application for Writing from Source Texts, *Hypertext '87 Papers*, 11/1987, pp. 121-141
- New Re:Mark 2.0, Ambia Corp., <http://www.ambia.com/remark.htm>, 1996
- New Version of Re:Mark Now Available, Ambia Corp., <http://www.ambia.com/press/remark20.htm>, 11/1996
- New Version of Re:mark Now Available, Re:Mark 2.0 Press Release, Ambia Corp., 11/1996, pp. 1-2
- Open eBook Forum, Open eBook.TM. Publication Structure 1.0, 9/1999, <http://www.ebxwg.org/oebps/oebps1.0/download/oeb1-oebps.htm>.
- Patrick, Industrial-Strength PDF Tools, *Adobe Magazine*, 2/1997, available at <http://www.ambia.com/compose-review.htm>

- pdfmark Reference Manual, Adobe Developer Support, Technical Note #5150, Adobe Systems Inc., 10/1996, pp. 1-54
- pdfmark Reference Manual, Adobe Systems Inc., 10/1996, pp. 1-54.
- Price et al., XLibris: The Active Reading Machine, Proceedings of CHI 98, 1998, pp. 1-2.
- Re:Mark Markup and Review of Adobe® Acrobat® software, Ambia Corp. brochure (undated)
- Re:mark® Markup and Review for Adobe® Acrobat® Software Version 2.0
- Re:mark® Markup and Review for Adobe® Acrobat® Software Version 2.0 User's Guide, Ambia Corp., 7/1997, pp. 1-157,
- Remarks, Ambia Corp., <http://www.ambia.com/remarks/current.htm>, 12/1996
- Remde et al., SuperBook: An automatic tool for information exploration - hypertext? *Proceedings of the ACM conference on Hypertext*, 11/1987, pp. 175-188.
- Render Annotations, Ambia Corp., <http://www.ambia.com/render.htm>, 1996
- Rocket eBook Using the Rocket eBook; 1999, <http://www.rocketbook.com/Products/Faq/using.html>.
- Roscheisen et al., Beyond Browsing: Shared Comments, SOAPs, Trails, and On-Line Communities, *Computer Networks and ISDN Sys* 27, 5/1995, pp. 739-749
- Roscheisen et al., Interaction Design for Shared WWW Annotations, *CHI '95 Mosaic of Creativity*, 5/1995, pp. 328-329
- Schickler et al., Pan-Browser Support for Annotations and Other Meta-Information on the WWW, *Computer Networks and ISDN Sys.* 28, 1996, pp. 1063-1074
- Schilit et al., Beyond Paper: Supporting Active Reading with Free Form Digital Ink Annotations, *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1998, pp. 1-8.
- Schilit et al., Digital Library Information Appliances, *Proceedings of Digital Libraries '98 – Third ACM Conference on Digital Libraries*, Pittsburg, PA, 6/1998, pp. 1-10
- Set Colors, Ambia Corp., <http://www.ambia.com/color-prop.htm>, 1996
- Set Line Width, Ambia Corp., <http://www.ambia.com/line-prop.htm>, 1996
- Smith, DynaText: An Electronic Publishing System, *Computers and the Humanities* 27, 1993, pp. 415-420

- SOFTBOOK™ Press, The Leader in Internet-Enabled Document Distribution and Reading Systems; 1999, <http://www.softbook.com/consumer/reader.asp>.
- Software Developer's Kit, Ambia Corp., <http://www.ambia.com/sdk.htm>, 1996
- Software Partners, Inc. – Adobe Acrobat Products and Services, <http://www.buckaroo.com>, 10/1996
- Supports Many Annotation Types, Ambia Corp., <http://www.ambia.com/remarksample.htm>, 1996
- The Leading developer of Adobe Acrobat software plug-in products and services, Ambia Corp., <http://www.ambia.com/ambia.htm>, 12/1996
- Troubleshooting Online Guide, Adobe Systems Inc., 1996
- U.S. Patent No. 4,424,575 to Clarke et al. (1/1984)
- U.S. Patent No. 5,063,600 to Norwood (11/1991)
- U.S. Patent No. 5,146,552 to Cassorla et al. (9/1992)
- U.S. Patent No. 5,220,658 to Kerr et al. (6/1993)
- U.S. Patent No. 5,231,578 to Levin et al. (7/1993)
- U.S. Patent No. 5,231,698 to Forcier (7/1993)
- U.S. Patent No. 5,239,466 to Morgan et al. (8/1993)
- U.S. Patent No. 5,253,362 to Nolan et al. (10/1993)
- U.S. Patent No. 5,347,295 to Agulnick et al. (9/1994)
- U.S. Patent No. 5,434,965 to Matheny et al. (7/1995)
- U.S. Patent No. 5,440,678 to Eisen (8/1995)
- U.S. Patent No. 5,442,793 to Christian et al. (8/1995)
- U.S. Patent No. 5,471,568 to Webb et al. (11/1995)
- U.S. Patent No. 5,473,742 to Polyakov et al. (12/1995)
- U.S. Patent No. 5,522,025 to Rosenstein (5/1996)
- U.S. Patent No. 5,557,722 to DeRose et al. (9/1996)

- U.S. Patent No. 5,559,942 to Gough et al. (9/1996)
- U.S. Patent No. 5,572,643 to Judson (11/1996)
- U.S. Patent No. 5,581,682 to Anderson et al. (12/1996)
- U.S. Patent No. 5,596,700 to Darnell et al. (1/1997)
- U.S. Patent No. 5,600,775 to King et al. (2/1997)
- U.S. Patent No. 5,615,325 to Peden (3/1997)
- U.S. Patent No. 5,623,591 to Cseri (4/1997)
- U.S. Patent No. 5,625,833 to Levine et al. (4/1997)
- U.S. Patent No. 5,630,125 to Zellweger (5/1997)
- U.S. Patent No. 5,632,022 to Warren et al. (5/1997)
- U.S. Patent No. 5,663,748 to Huffman (9/1997)
- U.S. Patent No. 5,678,039 to Hinks et al. (10/1997)
- U.S. Patent No. 5,680,636 to Levine et al. (10/1997)
- U.S. Patent No. 5,687,331 to Volk et al. (11/1997)
- U.S. Patent No. 5,689,717 to Pritt (11/1997)
- U.S. Patent No. 5,717,860 to Graber et al. (2/1998)
- U.S. Patent No. 5,717,879 to Moran (2/1998)
- U.S. Patent No. 5,719,595 to Hoddie et al. (2/1998)
- U.S. Patent No. 5,721,850 to Farry et al. (2/1998)
- U.S. Patent No. 5,727,129 to Barrett et al. (3/1998)
- U.S. Patent No. 5,729,687 to Rothrock et al. (3/1998)
- U.S. Patent No. 5,731,813 to O'Rourke et al. (3/1998)
- U.S. Patent No. 5,737,599 to Rowe et al. (4/1998)
- U.S. Patent No. 5,745,116 to Pisutha-Arnond (4/1998)

- U.S. Patent No. 5,790,818 to Martin (8/1998)
- U.S. Patent No. 5,799,268 to Boguraev (8/1998)
- U.S. Patent No. 5,801,687 to Peterson et al. (9/1998)
- U.S. Patent No. 5,819,301 to Rowe et al. (10/1998)
- U.S. Patent No. 5,821,925 to Casey et al. (10/1998)
- U.S. Patent No. 5,822,720 to Bookman et al. (10/1998)
- U.S. Patent No. 5,832,263 to Hansen et al. (11/1998)
- U.S. Patent No. 5,835,092 to Boudreau et al. (11/1998)
- U.S. Patent No. 5,838,313 to Hou et al. (11/1998)
- U.S. Patent No. 5,838,914 to Carleton et al. (11/1998)
- U.S. Patent No. 5,845,262 to Nozue et al. (12/1998)
- U.S. Patent No. 5,854,630 to Nielsen (12/1998)
- U.S. Patent No. 5,860,074 to Rowe et al. (1/1999)
- U.S. Patent No. 5,862,395 to Bier (1/1999)
- U.S. Patent No. 5,877,757 to Baldwin et al. (3/1999)
- U.S. Patent No. 5,893,126 to Drews et al. (4/1999)
- U.S. Patent No. 5,893,132 to Huffman et al. (4/1999)
- U.S. Patent No. 5,918,236 to Wical (6/1999)
- U.S. Patent No. 5,920,694 to Carleton et al. (7/1999)
- U.S. Patent No. 5,931,912 to Wu et al. (8/1999)
- U.S. Patent No. 5,933,139 to Feigner et al. (8/1999)
- U.S. Patent No. 5,933,140 to Strahorn et al. (8/1999)
- U.S. Patent No. 5,937,416 to Menzel (8/1999)
- U.S. Patent No. 5,938,724 to Pommier et al. (8/1999)

- U.S. Patent No. 5,940,080 to Ruehle et al. (8/1999)
- U.S. Patent No. 5,948,040 to DeLorme et al. (9/1999)
- U.S. Patent No. 5,950,214 to Rivette et al. (9/1999)
- U.S. Patent No. 5,956,034 to Sachs et al. (9/1999)
- U.S. Patent No. 5,978,818 to Lin (11/1999)
- U.S. Patent No. 5,986,665 to Wrey et al. (11/1999)
- U.S. Patent No. 5802516 to Shwartz et al. (9/1998)
- U.S. Patent No. 6,011,537 to Slotznick (1/2000)
- U.S. Patent No. 6,012,055 to Campbell et al. (1/2000)
- U.S. Patent No. 6,018,334 to Eckerberg et al. (1/2000)
- U.S. Patent No. 6,018,342 to Bristor (1/2000)
- U.S. Patent No. 6,018,344 to Harada et al. (1/2000)
- U.S. Patent No. 6,018,742 to Herbert, III (1/2000)
- U.S. Patent No. 6,034,689 to White et al. (3/2000)
- U.S. Patent No. 6,035,330 to Astiz et al. (3/2000)
- U.S. Patent No. 6,037,934 to Himmel et al. (3/2000)
- U.S. Patent No. 6,038,598 to Danneels (3/2000)
- U.S. Patent No. 6,041,335 to Merritt et al. (3/2000)
- U.S. Patent No. 6,049,812 to Bertram et al. (4/2000)
- U.S. Patent No. 6,052,514 to Gill et al. (4/2000)
- U.S. Patent No. 6,054,990 to Tran (4/2000)
- U.S. Patent No. 6,072,490 to Bates et al. (6/2000)
- U.S. Patent No. 6,076,917 to Wen (6/2000)
- U.S. Patent No. 6,081,829 to Sidana (6/2000)

- U.S. Patent No. 6,088,481 to Okamoto et al. (7/2000)
- U.S. Patent No. 6,088,709 to Watanabe (7/2000)
- U.S. Patent No. 6,091,930 to Mortimer et al. (7/2000)
- U.S. Patent No. 6,092,081 to Alpert et al. (7/2000)
- U.S. Patent No. 6,094,197 to Buxton et al. (7/2000)
- U.S. Patent No. 6,105,044 to DeRose et al. (8/2000)
- U.S. Patent No. 6,118,437 to Fleck et al. (9/2000)
- U.S. Patent No. 6,122,649 to Kanerva et al. (9/2000)
- U.S. Patent No. 6,128,007 to Seybold (10/2000)
- U.S. Patent No. 6,133,925 to Jaremko et al. (10/2000)
- U.S. Patent No. 6,144,991 to England (11/2000)
- U.S. Patent No. 6,151,021 to Berquist et al. (11/2000)
- U.S. Patent No. 6,154,757 to Krause et al. (11/2000)
- U.S. Patent No. 6,154,757 to Nolan et al. (10/1993)
- U.S. Patent No. 6,154,771 to Rangan et al.(11/2000)
- U.S. Patent No. 6,157,381 to Bates et al. (12/2000)
- U.S. Patent No. 6,163,778 to Fogg et al. (12/2000)
- U.S. Patent No. 6,181,344 to Tarpenning et al. (1/2001)
- U.S. Patent No. 6,184,886 to Bates et al. (2/2001)
- U.S. Patent No. 6,195,679 to Bauersfeld et al. (2/2001)
- U.S. Patent No. 6,195,694 to Chen et al. (2/2001)
- U.S. Patent No. 6,199,082 to Ferrel et al. (3/2001)
- U.S. Patent No. 6,205,455 to Umen et al. (3/2001)
- U.S. Patent No. 6,211,871 to Himmel et al. (4/2001)

- U.S. Patent No. 6,212,532 to Johnson et al. (4/2001)
- U.S. Patent No. 6,230,169 to Nagae (5/2001)
- U.S. Patent No. 6,230,171 to Pacifici et al. (5/2001)
- U.S. Patent No. 6,239,792 to Yanagisawa et al. (5/2001)
- U.S. Patent No. 6,243,071 to Shwarts et al. (6/2001)
- U.S. Patent No. 6,243,091 to Berstis (6/2001)
- U.S. Patent No. 6,243,722 to Day et al. (6/2001)
- U.S. Patent No. 6,259,446 to Matheny et al. (7/2001)
- U.S. Patent No. 6,262,719 to Bi et al. (7/2001)
- U.S. Patent No. 6,266,772 to Suzuki (7/2001)
- U.S. Patent No. 6,271,840 to Finseth et al.(8/2001)
- U.S. Patent No. 6,279,005 to Zellweger (8/2001)
- U.S. Patent No. 6,279,014 to Schilit et al. (8/2001)
- U.S. Patent No. 6,289,362 to Van Der Meer (9/2001)
- U.S. Patent No. 6,289,460 to Hajmiragha (9/2001)
- U.S. Patent No. 6,301,590 to Siow et al. (10/2001)
- U.S. Patent No. 6,321,244 to Liu et al. (11/2001)
- U.S. Patent No. 6,331,867 to Eberhard et al. (12/2001)
- U.S. Patent No. 6,335,727 to Morishita et al. (1/2002)
- U.S. Patent No. 6,340,967 to Maxted (1/2002)
- U.S. Patent No. 6,342,906 to Kumar et al. (1/2002)
- U.S. Patent No. 6,356,287 to Ruberry et al. (3/2002)
- U.S. Patent No. 6,369,811 to Graham et al. (4/2002)
- U.S. Patent No. 6,370,497 to Knowles (4/2002)

- U.S. Patent No. 6,377,983 to Cohen et al. (4/2002)
- U.S. Patent No. 6,389,434 to Rivette et al. (5/2002)
- U.S. Patent No. 6,393,422 to Wone (5/2002)
- U.S. Patent No. 6,397,264 to Stasnick et al. (5/2002)
- U.S. Patent No. 6,405,221 to Levine et al. (6/2002)
- U.S. Patent No. 6,421,065 to Walden et al. (7/2002)
- U.S. Patent No. 6,425,525 to Swaminathan et al. (7/2002)
- U.S. Patent No. 6,437,793 to Kaasila (8/2002)
- U.S. Patent No. 6,438,564 to Morton et al. (8/2002)
- U.S. Patent No. 6,446,110 to Lection et al.(9/2002)
- U.S. Patent No. 6,457,013 to Saxton et al. (9/2002)
- U.S. Patent No. 6,460,058 to Koppolu et al. (10/2002)
- U.S. Patent No. 6,484,156 to Gupta et al. (11/2002)
- U.S. Patent No. 6,486,895 to Robertson et al. (11/2002)
- U.S. Patent No. 6,490,603 to Keenan et al. (12/2002)
- U.S. Patent No. 6,529,920 to Arons et al. (3/2003)
- U.S. Patent No. 6,535,294 to Arledge, Jr. et al. (3/2003)
- U.S. Patent No. 6,539,370 to Chang et al. (3/2003)
- U.S. Patent No. 6,546,405 to Gupta et al. (4/2003)
- U.S. Patent No. 6,549,220 to Hsu et al. (4/2003)
- U.S. Patent No. 6,551,357 to Madduri (4/2003)
- U.S. Patent No. 6,560,621 to Barile (5/2003)
- U.S. Patent No. 6,571,211 to Dwyer et al. (5/2003)
- U.S. Patent No. 6,571,295 to Sidana (5/2003)

- U.S. Patent No. 6,585,776 to Bates et al. (7/2003)
- U.S. Patent No. 6,647,534 to Graham (11/2003)
- U.S. Patent No. 6,697,997 to Fujimura (2/2004)
- U.S. Patent No. 6,760,884 to Vertelney et al. (7/2004)
- U.S. Patent No. 6,865,713 to Bates et al. (3/2005)
- U.S. Patent No. 7,000,186 to Gropper et al. (2/2006)
- U.S. Patent No. 7,107,450 to Yu (9/2006)
- U.S. Patent No. 7,228,492 to Graham (6/2007)
- U.S. Patent No. 7,432,938 to Reuter et al. (10/2008)
- U.S. Patent No. 7,574,649 to Safars et al. (8/2009)
- U.S. Patent No. 7,574,649 to Safars et al. (8/2009)
- U.S. Patent No. RE 34,476 to Norwood (12/1993)
- U.S. Patent Pub. 2001/0031128 to Manbeck (10/2001)
- U.S. Patent Pub. 2002/0097261 to Gottfurcht et al. (7/2002)
- U.S. Patent Pub. 2002/0099552 to Rubin et al. (7/2002)
- Undo Annotation Rendering, Ambia Corp., <http://www.ambia.com/unrender.htm>, 1996
- UNIX Daemon Administration Guide Adobe Acrobat version 3.0, Adobe Systems Inc., 1996, pp. 1-23
- WO 87/01481 (A1) to Stein et al. (3/1987)
- WO 96/20908 (7/1996)
- WO 98/09446 (A2) to Ho (3/1998)
- B. Name and address of any person who may be relied upon as the prior inventor or as having prior knowledge of or as having previously used or offered for sale:**
- Individuals formerly or currently associated with Adobe Systems Inc. (address unknown)

- Individuals formerly associated with Image Solutions, Inc., Software Partners, Inc., Ambia Corp., Infodata Systems, Inc., or Computer Sciences Corp (address unknown)

Dated: October 7, 2011

Respectfully submitted,

/s/ Marcia H. Sundeen

Richard DeLucia
Elizabeth Gardner
John Kenny
Paul Richter
Charles Weiss
Antony Pfeffer
Kenyon & Kenyon LLP
One Broadway
New York, NY 10004-1050
Telephone: (212) 425-7200
Fax: (212) 425-5288

Marcia H. Sundeen
Jeffrey S. Gerchick
Aimee N. Soucie
Kenyon & Kenyon LLP
1500 K Street, N.W.
Washington, DC 20005
Tel: (202) 220-4200
Fax: (202) 220-4201

*Counsel for Respondents Barnes & Noble, Inc.
and barnesandnoble.com LLC*

**Certain Handheld Electronic Computing Devices,
Related Software and Components Thereof**

337-TA-769

CERTIFICATE OF SERVICE

I hereby certify that copies of the foregoing **BARNES & NOBLE, INC.'S AND BARNESANDNOBLE.COM LLC'S NOTICE OF PRIOR ART** were served upon the following parties as indicated on this 7th day of October, 2011.

James R. Holbein, Secretary
U.S. International Trade Commission
500 E Street, S.W., Room 112
Washington, D.C. 20436

- ☐ Via Hand Delivery
☐ Via Overnight Federal Express Delivery
☐ Via First Class Mail
☐ Via Facsimile
☒ Via Electronic Service

Honorable Theodore R. Essex
U.S. International Trade Commission
500 E Street, S.W., Room 317
Washington, D.C. 20436
Email: tamara.foley@usitc.gov

- ☒ Via Hand Delivery
☐ Via Overnight Federal Express Delivery
☐ Via First Class Mail
☐ Via Facsimile
☒ Via Electronic Mail

Jeffrey Hsu
Office of Unfair Import Investigations
U.S. International Trade Commission
500 E Street, S.W.,
Washington, D.C. 20436
Email: jeffrey.hsu@usitc.gov

- ☐ Via Hand Delivery
☐ Via Overnight Federal Express Delivery
☐ Via First Class Mail
☐ Via Facsimile
☒ Via Electronic Mail

Counsel for Complainant Microsoft Corporation

V. James Adduci
Adduci, Mastriani & Schaumberg LLP
1200 Seventeenth Street, NW 5th Floor
Washington, DC 20036
Email: MSFT-2@adduci.com, MSFT-L14External@woodcock.com, MSFT_BN@orrick.com

- ☐ Via Hand Delivery
☐ Via Overnight Federal Express Delivery
☐ Via First Class Mail
☐ Via Facsimile
☒ Via Electronic Mail

*Counsel for Respondent Hon Hai Precision Industry Co., Ltd.,
Foxconn Electronics, Inc., Foxconn International Holdings Ltd.,
and Foxconn Precision Component (Shenzhen) Co., Ltd.*

Edward M. Lebow
Haynes and Boone LLP
1615 L Street, NW Suite 800
Washington, DC 20036
Email: HonHai769@haynesboone.com

- ☐ Via Hand Delivery
☐ Via Overnight Federal Express Delivery
☐ Via International First Class Mail
☐ Via Facsimile
☒ Via Electronic Mail

/s/ Hanna Cha

Hanna Cha
Legal Assistant